NASA TECHNICAL MEMORANDUM

NASA TM X-2805

NASA TM X-2805

A COMPUTER PROGRAM TO CALCULATE RADIATING VISCOUS STAGNATION STREAMLINE FLOW WITH STRONG BLOWING

by G. Louis Smith and L. Bernard Garrett

Langley Research Center
Hampton, Va. 23665

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • DECEMBER 1973

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA TM X-2805		

4. Title and Subtitle	5. Report Date
A COMPUTER PROGRAM TO CALCULATE RADIATING VISCOUS STAGNATION STREAMLINE FLOW WITH STRONG BLOWING	December 1973
	6. Performing Organization Code
7. Author(s)	8. Performing Organization Report No.
G. Louis Smith and L. Bernard Garrett	L-8918
	10. Work Unit No.
9. Performing Organization Name and Address	502-27-01-01
NASA Langley Research Center Hampton, Va. 23665	11. Contract or Grant No.
	13. Type of Report and Period Covered
12. Sponsoring Agency Name and Address	Technical Memorandum
National Aeronautics and Space Administration Washington, D.C. 20546	14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

A computer program (program LEE) has been developed to calculate the fully coupled solution of the radiating viscous stagnation streamline flow with strong blowing. Program LEE was developed for the study reported in NASA TR R-388. The present report describes the digital computer program, including FORTRAN IV listing, flow charts, instructions for the user, and a test case with input and output. Program LEE is available through COSMIC.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Radiative heat transfer Shock layer, viscous Stagnation flow Equilibrium flow Mass transfer	Unclassified – Unlimited

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	81	Domestic, $3.75 Foreign, $6.25

*For sale by the National Technical Information Service, Springfield, Virginia 22151

CONTENTS

A COMPUTER PROGRAM TO CALCULATE RADIATING VISCOUS STAGNATION STREAMLINE FLOW WITH STRONG BLOWING

By G. Louis Smith and L. Bernard Garrett
Langley Research Center

SUMMARY

A computer program (program LEE) has been developed to calculate the fully coupled solution of the radiating viscous stagnation streamline flow with strong blowing. Program LEE was developed for the study reported in NASA TR R-388. The present report describes the digital computer program, including FORTRAN IV listing, flow charts, instructions for the user, and a test case with input and output. Program LEE is available through COSMIC.

INTRODUCTION

An implicit finite difference solution to the viscous shock-layer stagnation streamline flow with radiation and strong blowing is presented in references 1 and 2. Program LEE (Langley computer program A3899) was developed to make the calculations reported in references 1 and 2. A detailed radiative transport code (RATRAP) that accounts for the important radiative exchange processes for gaseous mixtures is used (ref. 3). Chemical equilibrium of the flow is assumed, and the equilibrium composition of the air-injection products mixture is computed by a free-energy minimization routine (FEMP) within RATRAP.

The program may be used for the stagnation streamline of a two-dimensional body, an axisymmetric body, or a three-dimensional body with two orthogonal planes of symmetry. The difference between these cases is mainly in the continuity equation. For the three-dimensional case, there are two components of velocity around the body to consider, but the momentum equation is the same for both components.

A description of the computer program is presented in this paper. Included are program listing, flow charts, instructions for the user, and a test case with input and output listings. Copies of the program LEE computer deck are available through COSMIC (Computer Software Management and Information Center at Barrow Hall, University of Georgia, Athens, Georgia).

SYMBOLS

a	tangential velocity gradient, $\frac{\partial u}{\partial x}$
b	tangential velocity gradient, $\frac{\partial w}{\partial z}$
c_p	specific heat at constant pressure
D_{12}	binary diffusion coefficient
H	specific total enthalpy
h	specific static enthalpy
I	index for nodal point
K	body curvature
K_z	body curvature in z-direction for nonaxisymmetric body
k	thermal conductivity
N	number of node points in finite difference solution
N_{Pr}	Prandtl number
N_{Re}	Reynolds number
p	static pressure
p'_∞	free-stream static pressure, dynes/cm^2 (where 1 dyne = 10^{-5} newton)
$q_{R,y}$	net radiative heat flux in y-direction
R'_b	radius of body, cm

r radius measured from axis of symmetry of body (see fig. 1)

U'_∞ free-stream velocity, cm/sec

u component of velocity parallel to surface in x-direction

v component of velocity normal to surface

w component of velocity parallel to surface in z-direction

x distance measured along body surface (see fig. 1)

y distance measured normal to body surface (see fig. 1)

z distance measured along body surface in plane normal to plane in which x is measured

α_i mass fraction of ith chemical species

$\tilde{\alpha}_F$ *mass fraction of foreign or ablation products*

$\beta = -\left(\frac{\partial^2 p}{\partial x^2}\right)_{x=0}$

$\beta_z = -\left(\frac{\partial^2 p}{\partial z^2}\right)_{z=0}$

$\hat{\delta}$ transformed shock-layer thickness, $\int_0^{y_s} \rho(y)\,dy$

ϵ error criterion for density

η transformed coordinate normal to body surface (see fig. 2), $\frac{1}{\hat{\delta}}\int_0^{y} \rho(y)\,dy$

κ scale factor, $1 + Ky$

κ_i dummy scale factor

κ_z scale factor for curvature in z-direction, $1 + K_z y$

μ viscosity

ρ density

Subscripts:

C carbon

H hydrogen

N nitrogen

O oxygen

n nth nodal point; n = 1 at the body (η = 0) and n = N at the shock (η = 1)

s shock

∞ free-stream conditions

Superscript:

i ith iteration

Unprimed symbols are nondimensional. All dimensional symbols are primed.

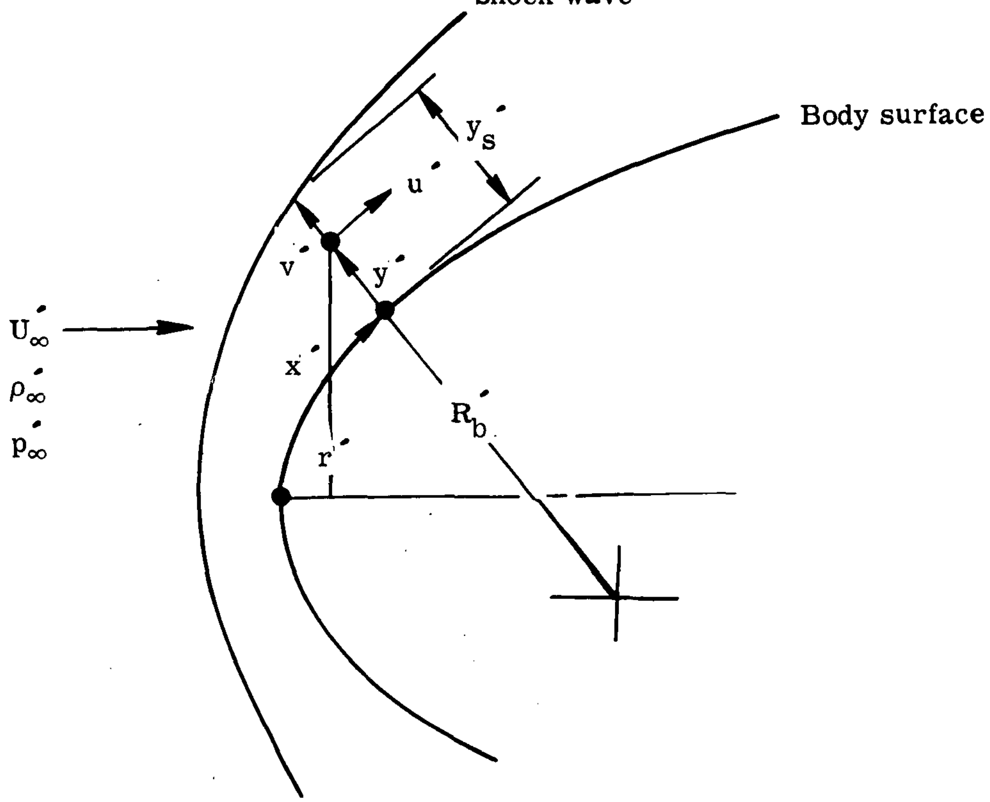


Figure 1. - Flow-field coordinate system.

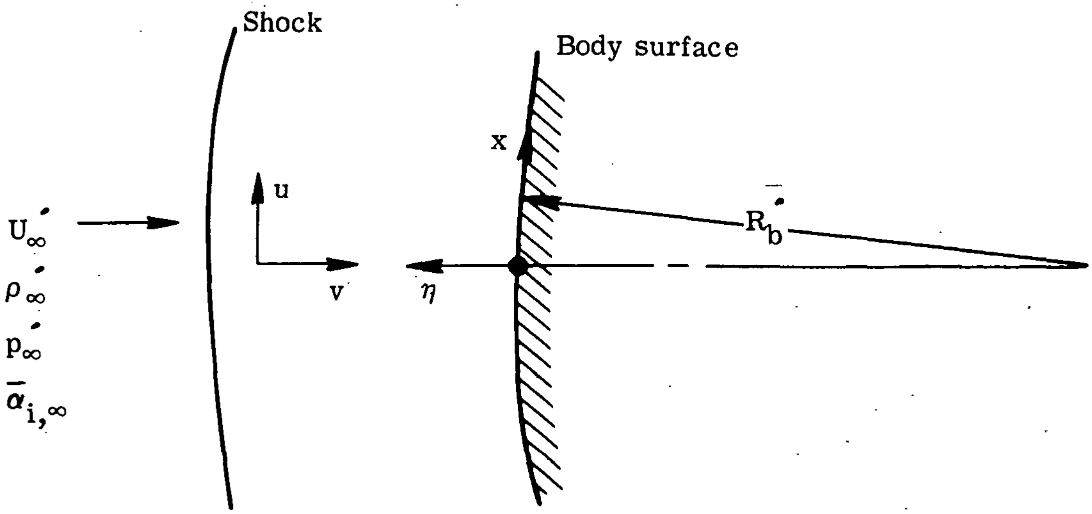


Figure 2. - Flow-field coordinate system in the transformed coordinates.

PROBLEM DESCRIPTION

The equations which govern radiating viscous flow along the stagnation streamline of a blunt body in nondimensional form are as follows (ref. 1 or 2):

Continuity:

$$\frac{d}{dy}(\kappa\rho v) = -\kappa\rho a \qquad \text{(Two dimensional)} \qquad (1a)$$

$$\frac{d\left(\kappa^2\rho v\right)}{dy} = -2\,\kappa\,\rho a \qquad \text{(Axisymmetric)} \qquad (1b)$$

$$\frac{d\left(\kappa\kappa_z\rho v\right)}{dy} = -\kappa\rho a - \kappa_z\rho b \qquad \text{(Three dimensional)} \qquad (1c)$$

x-momentum:

$$-\frac{\kappa}{N_{Re}}\frac{d}{dy}\left(\mu\frac{da}{dy}\right) + \kappa\rho v\frac{da}{dy} + \rho a^2 + K\rho av = \beta \qquad (2a)$$

z-momentum:

$$-\frac{\kappa_z}{N_{Re}}\frac{d}{dy}\left(\mu\frac{db}{dy}\right) + \kappa_z\rho v\frac{db}{dy} + \rho b^2 + K_z\rho bv = \beta_z \qquad (2b)$$

y-momentum:

$$\rho v\frac{dv}{dy} = -\frac{dp}{dy} \qquad (3)$$

Energy:

$$\frac{1}{N_{Re}}\frac{d}{dy}\left(\frac{\kappa\kappa_i\mu}{N_{Pr}}\frac{dH}{dy}\right) - \kappa\kappa_i\rho v\frac{dH}{dy} = \frac{1}{N_{Re}}\frac{d}{dy}\left(\frac{\kappa\kappa_i}{N_{Pr}}v\frac{dv}{dy}\right) + \frac{d}{dy}\left(\kappa\kappa_i q_{R,y}\right) \qquad (4)$$

Diffusion:

$$\frac{d}{dy}\left(\kappa\kappa_i \rho D_{12} \frac{d\bar{a}_F}{dy}\right) - \kappa\kappa_i \rho v \frac{d\bar{a}_F}{dy} = 0 \tag{5}$$

These equations are written in the coordinate system shown in figure 1. In these equations, $\kappa_i = 1$ for the two-dimensional case, $\kappa_i = \kappa$ for the axisymmetric case, and $\kappa_i = \kappa_z$ for the three-dimensional case.

For the case of a three-dimensional body with two orthogonal planes of symmetry x is the distance from the stagnation point around the body in one plane of symmetry and z is the distance measured in the other plane of symmetry. There are two corresponding orthogonal components of velocity u and w parallel to the body surface. The stagnation streamline equations, which are the limit forms of the fluid-flow equations, involve the terms $a = \frac{\partial u}{\partial x}$ and $b = \frac{\partial w}{\partial z}$. For two-dimensional flow, b and z-curvature vanish. For axisymmetric flow, b is identical with a. The quantities have been nondimensionalized as follows (where primes denote dimensional quantities):

$$\left.\begin{array}{lll}
x, y, r = \dfrac{x', y', r'}{R'_b} & u, v, w = \dfrac{u', v', w'}{U'_\infty} & \\
K = K'R'_b & \rho = \dfrac{\rho'}{\rho'_\infty} & p = \dfrac{p'}{\rho'_\infty \left(U'_\infty\right)^2} \\
h, H = \dfrac{h', H'}{\left(U'_\infty\right)^2} & q_{R,y} = \dfrac{q'_{R,y}}{\rho'_\infty \left(U'_\infty\right)^3} & k = \dfrac{k'}{k'_s} \\
D_{12} = \dfrac{D'_{12}}{U'_\infty R'_b} & \mu = \dfrac{\mu'}{\mu'_s} & a = \dfrac{R'_b}{U'_\infty} \dfrac{\partial u'}{\partial x'} \\
N_{Re} = \dfrac{\rho'_\infty U'_\infty R'_b}{\mu'_s} & N_{Pr} = \dfrac{c'_p \mu'}{k'} & b = \dfrac{R'_b}{U'_\infty} \dfrac{\partial w'}{\partial z'}
\end{array}\right\} \tag{6}$$

The symbol R'_b which is used for nondimensionalization is not restricted to being the nose radius of curvature. For example, for a flat disk, R'_b can be the disk radius. The following restrictions are made on the problem:

(1) The gas is in local thermodynamic and chemical equilibrium.

(2) Diffusion is governed by a binary diffusion model.

(3) Radiative energy transport occurs within a one-dimensional, infinite, planar slab (tangent-slab approximation).

(4) The gas density must be high enough that the shock wave is thin relative to the shock layer.

The transform variable

$$\eta = \frac{1}{\hat{\delta}} \int_0^y \rho(y)\, dy \tag{7}$$

is defined where

$$\hat{\delta} = \int_0^{y_s} \rho(y)\, dy \tag{8}$$

It is seen that this transformation is defined such that the shock wave is fixed at $\eta = 1$. The transformation of equation (7) is applied to equations (1) to (5) so that they can be written, respectively, as follows:

Continuity:

$$\frac{d(\kappa \rho v)}{d\eta} = \hat{\delta} \kappa a \qquad \text{(Two dimensional)} \tag{9a}$$

$$\frac{d(\kappa^2 \rho v)}{d\eta} = 2\hat{\delta} \kappa a \qquad \text{(Axisymmetric)} \tag{9b}$$

$$\frac{d(\kappa \kappa_z \rho v)}{d\eta} = \hat{\delta} \kappa a = \hat{\delta} \kappa_z b \qquad \text{(Three dimensional)} \tag{9c}$$

x-momentum:

$$\frac{\kappa \rho}{N_{Re} \hat{\delta}^2} \frac{d}{d\eta}\left(\mu \rho \frac{da}{d\eta}\right) + \frac{\kappa \rho^2 v}{\hat{\delta}} \frac{da}{d\eta} - \rho a^2 + K \rho a v = -\beta \tag{10a}$$

z-momentum:

$$\frac{\kappa_z \rho'}{N_{Re}\hat{\delta}^2} \frac{d}{d\eta}\left(\mu\rho \frac{db}{d\eta}\right) + \frac{\kappa_z \rho^2 v}{\hat{\delta}} \frac{db}{d\eta} - \rho b^2 + K_z \rho b v = -\beta_z \tag{10b}$$

y-momentum:

$$\frac{dp}{d\eta} = -\rho v \frac{dv}{d\eta} \tag{11}$$

Energy:

$$\frac{d}{d\eta} \frac{\kappa\kappa_i \rho\mu}{N_{Pr}} \frac{dH}{d\eta} + \hat{\delta} N_{Re} \kappa\kappa_i \rho v \frac{dH}{d\eta} =$$

$$= \frac{d}{d\eta}\left(\frac{\kappa\kappa_i \mu}{N_{Pr}} \rho v \frac{dv}{d\eta}\right) + \hat{\delta} N_{Re} \frac{d}{d\eta}\left(\kappa\kappa_i q_{R,\eta}\right) \tag{12}$$

Diffusion:

$$\frac{d}{d\eta}\left(\kappa\kappa_i \rho^2 D_{12} \frac{d\bar{a}_F}{d\eta}\right) + \hat{\delta}\kappa\kappa_i \rho v \frac{d\bar{a}_F}{d\eta} = 0 \tag{13}$$

The user is cautioned that the normal velocity v has been redefined as negative in the positive y- or η-direction in the aforementioned equations. The purpose of this sign change is to make v positive across the shock layer, except within the layer of injected products. The flow-field coordinate system is shown in figure 2. The boundary condition at the wall for the diffusion equation has been generalized from that used in references 1 and 2 for strong blowing and is

$$D_{12} \frac{d\bar{a}_F}{dy} + v\bar{a}_F = v \qquad (\eta = 0)$$

This boundary condition is appropriate also for moderate blowing rates.

Equations (10), (12), and (13) are written in finite difference form by using a central-difference formula for second derivatives and windward differencing for first derivatives in convective terms. Equations (9) and (11) are integrated for $(\kappa\kappa_i \rho v)$ and p, respectively, subject to boundary conditions at the shock wave. Up to 100 node points may be used for the solution. These node points are evenly spaced in η.

PROGRAM ORGANIZATION

The governing equations (9) to (13) are solved by an iterative scheme as shown in figure 3 and are discussed in reference 2. This section of the report describes subprograms used to carry out the computation. A flow chart of each subprogram is included except where computation is straightforward. A full listing of program LEE is given in appendix A. The input is described in appendix B, and the output is described in appendix C. Langley library subroutines which are used are described in appendixes D, E, and F. A list of each subprogram and its description is given as follows:

Subprogram	Description
LEE	Initializes program, iterates density and flow equations, and prints results
GETREDY	Reads spectral data input for RATRAP
INITIAL	Initializes flow variables
RANHUG	Solves Rankine-Hugoniot relations for post-shock conditions
FOFXRO	Function used by RANHUG
UPDATE	Option of input for initial profiles
FLOW	Iterates flow equations to solution
CHEXRO	Computes density, compares with previous value
MASS	Solves continuity equation for v
Y MOMNTM	Solves y-momentum equation for p
SPECIES	Solves diffusion equation for $\bar{a}_F$ and elemental mass fractions

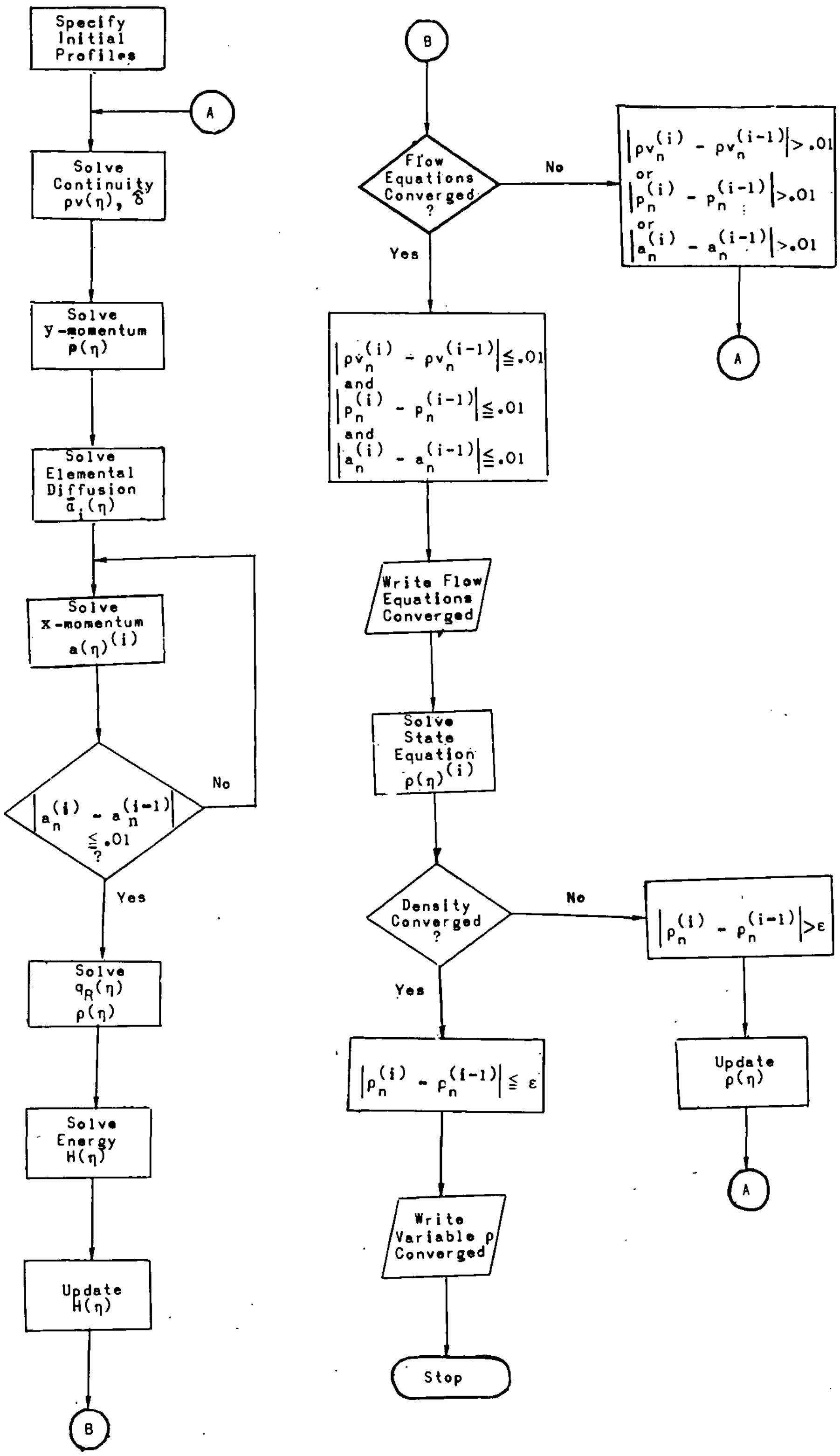


Figure 3. - Flow chart of the overall solution procedure.

Subprogram	Description
X MOMNTM	Solves x-momentum equation for a and, if necessary, for b
ENERGY	Solves energy equation for h and H
RADARAY	Calls RATRAP for radiation heat flux at NIC points and linearly interpolates for remainder of N points; also updates density
REFTEMP	Computes post-shock temperature
D2DP2	Computes $\beta = -\frac{\partial^2 p}{\partial x^2}$
DIMPROP	Computes dimensional p' and h', given nondimensional p and h
RHOSY	Equation of state, $\rho(p, h, \alpha_i)$
DIFCOEF	Computes binary diffusion coefficient
VISPRAN	Computes Prandtl number and nondimensional viscosity profiles, given p and h profiles
VISCOS	Computes dimensional viscosity using tables of reference 4
PRANDTL	Computes Prandtl number using tables of reference 4
DIFT	Given a profile $y_i(x_i)$, computes profile $\frac{dy}{dx}(x_i)$
TINT	Given a profile $y_i(x_i)$, computes profile $\int_0^{x_i} y(\xi)\, d\xi$, where ξ is a dummy variable
TRIDIAG	Solves Ax = D by Potter's method where A is a tridiagonal matrix and x and D are vectors

Subprogram	Description
SET	Sets one array equal to another
ITEST	Computes number of C(I) differing from B(I), where C and B are arrays, by amount exceeding a specified E
RATRAP	Radiation transport code (ref. 3)

The radiation transport code RATRAP, developed by K. H. Wilson (ref. 3), is used as a subroutine. It uses a number of subprograms which are listed as part of appendix A; these are all described in reference 3. Included among these is a free-energy minimization program (FEMP) which solves for the mass fractions of the individual species. The many spectral data required for RATRAP are read in by GETREDY at the beginning of the computations. Radiative flux is computed at each of a desired number N of evenly spaced points. The N is input in NAMELIST NAM4.

Langley library subroutines of the SCOPE 3.0 operating system which are used in the LEE program are ITR1, FTLUP, and DISCOT. These are described in appendixes D, E, and F, respectively.

In some of the flow charts, the operations "damp and clip" appear. "Damp" refers to computing a value for a given variable by weighting the value most recently computed with its value from the previous iteration, for each nodal point. For example, the value computed for density would be

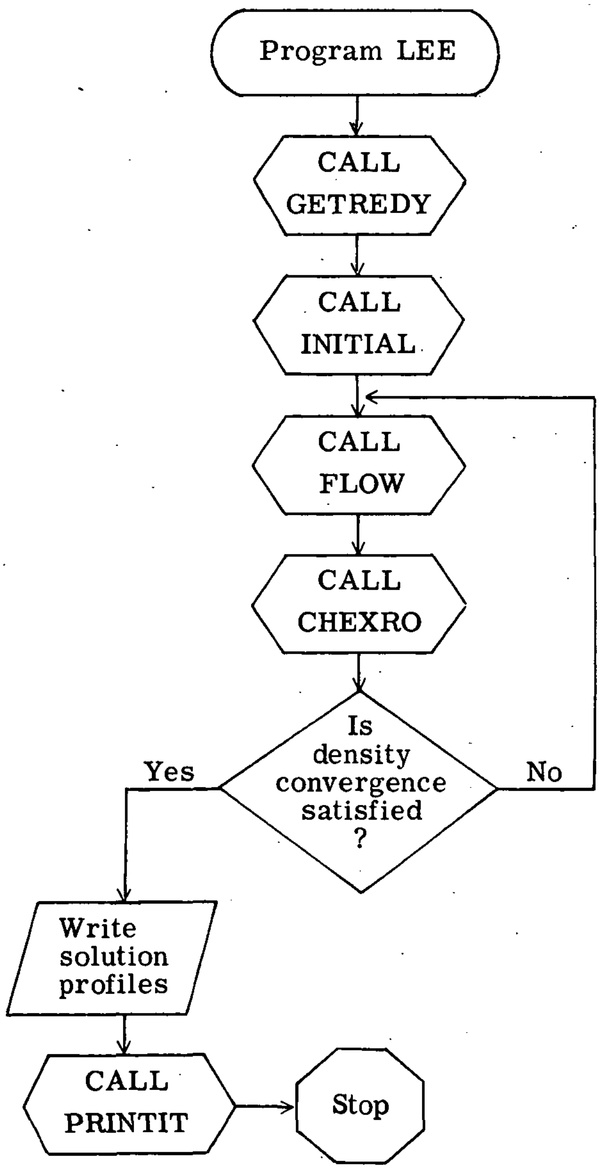
$$\rho_n = d_\rho \rho_n^{(i-1)} + (1 - d_\rho) \rho_n^{(i)}$$

where d_ρ is the damping factor for density. The purpose of damping is to suppress numerical oscillations from one iteration to the next, and thereby improve convergence. The selection of d_ρ is discussed in references 1 and 2. "Clip" refers to the requirement that a profile not go below or above some value. For example, an intermediate iteration may give an excessive radiation divergence term, thus causing an unrealistic dip in the enthalpy profile. This condition is alleviated by requiring that H(I) be at least as large as the wall value. When this adjustment is required, a message is printed.

PRIMARY SUBPROGRAMS

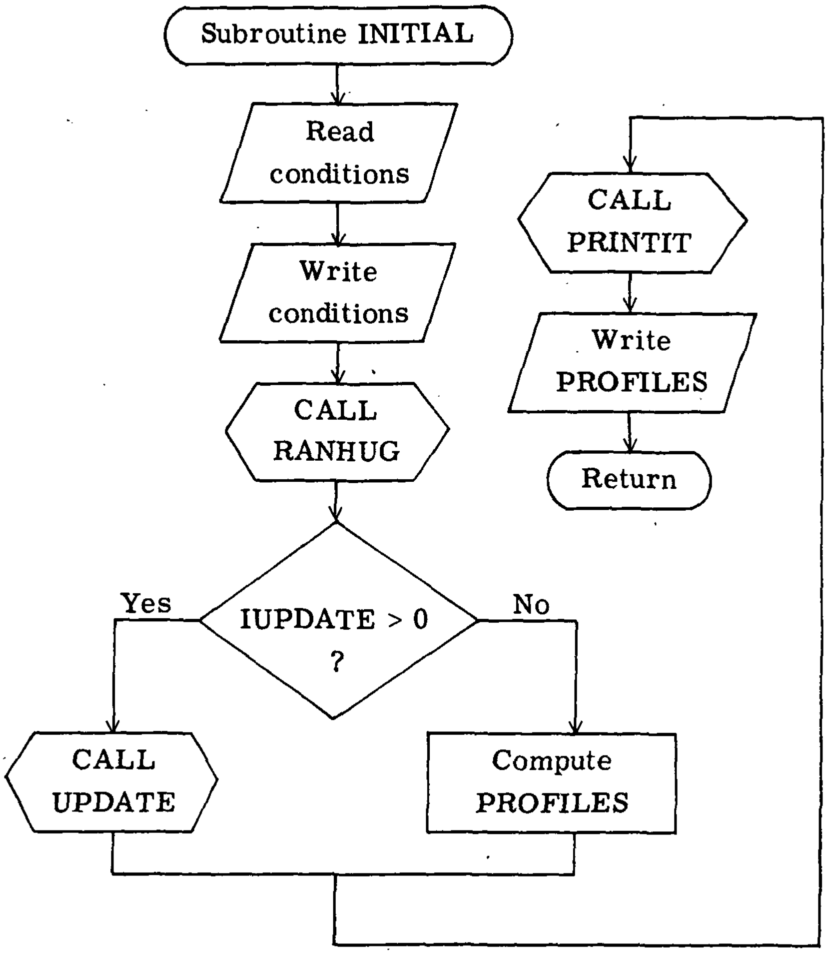
Program LEE

LEE is the main program. It calls subroutines GETREDY and INITIAL, which input data and initialize profiles, respectively. Then, LEE directs the solution of the fluid-flow equations and the density computation by successive iteration until convergence is achieved. The flow chart for LEE is given as follows:



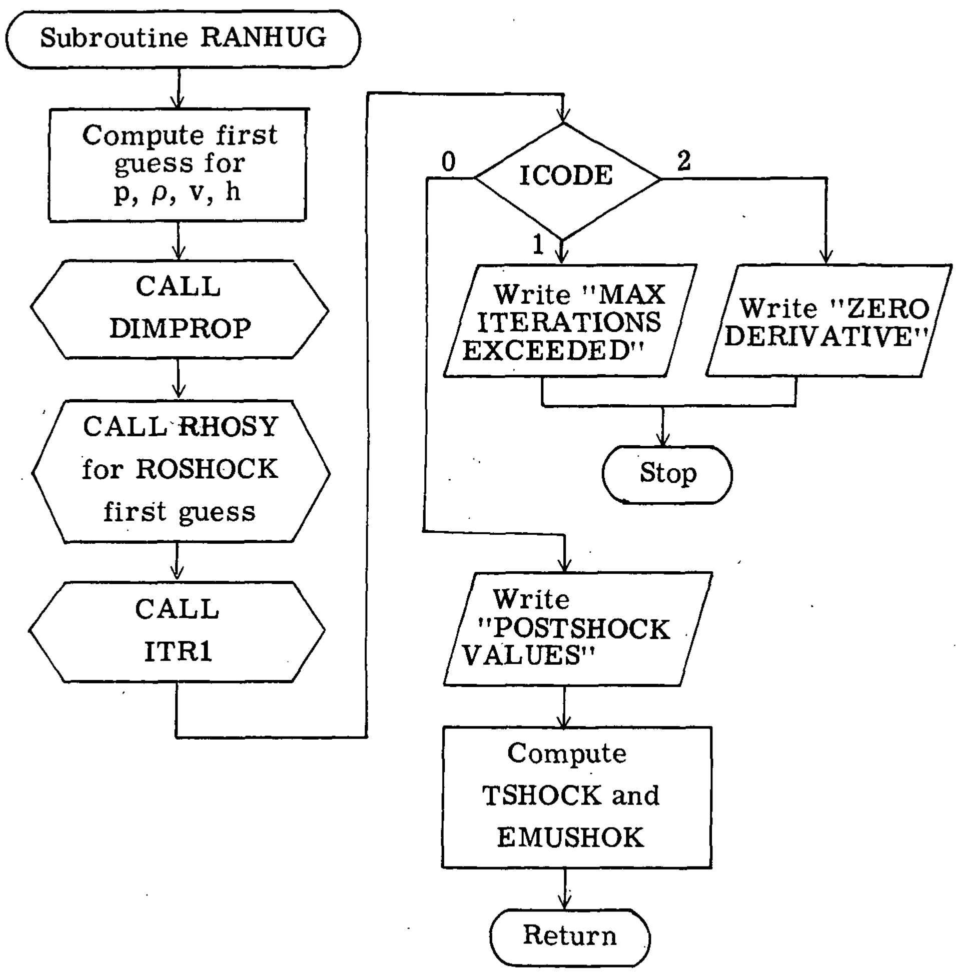
Subroutine INITIAL

The INITIAL subroutine performs those tasks which must be done before the interaction of the profiles can begin. The conditions of the problem are read in. Post-shock conditions are computed. Profiles of various flow quantities are initialized by one of two options. If IUPDATE $\leqq 0$, linear and second-order profiles are assumed initially across the shock layer. If IUPDATE > 0, the UPDATE subroutine is called to input profiles. The flow chart for INITIAL is given as follows:



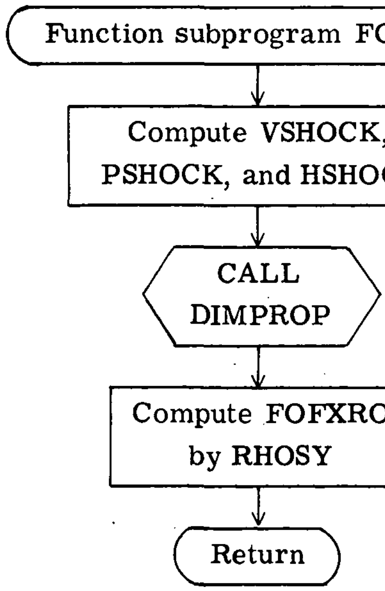
Subroutine RANHUG

The RANHUG subroutine computes the properties behind a normal shock wave. An initial guess of pressure and static enthalpy is made and an initial guess for density ROSHOCK is computed on this basis. Starting with this value, a Newton-Raphson subroutine ITR1 iterates to compute ROSHOCK to the desired accuracy. The ITR1 subroutine is described in appendix D. The flow chart for RANHUG is given as follows:



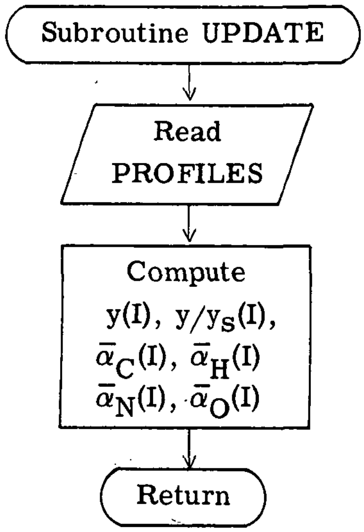
Function Subprogram FOFXRO

The FOFXRO function subprogram calls ITR1 to compute ROSHOCK. It uses the Rankine-Hugoniot equations and equation of state (as represented by RHOSY) to evaluate density ROSHOCK. The flow chart for FOFXRO is given as follows:



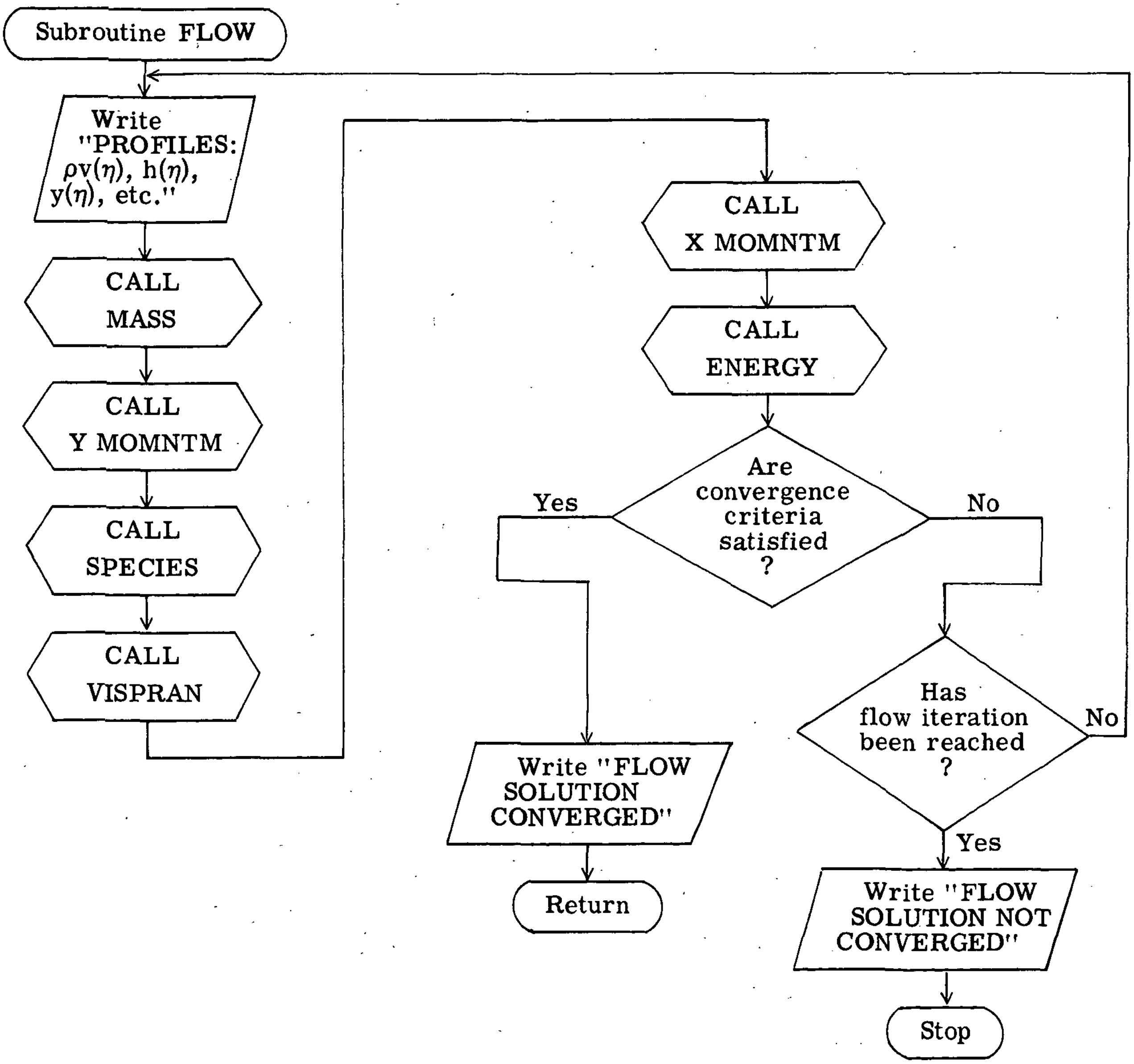
Subroutine UPDATE

The UPDATE subroutine reads NAMELISTs for the profiles required to begin the iteration for the solution, provided that IUPDATE >0. This option allows the user to begin the iteration with the best possible guess. The flow chart for UPDATE is given as follows:



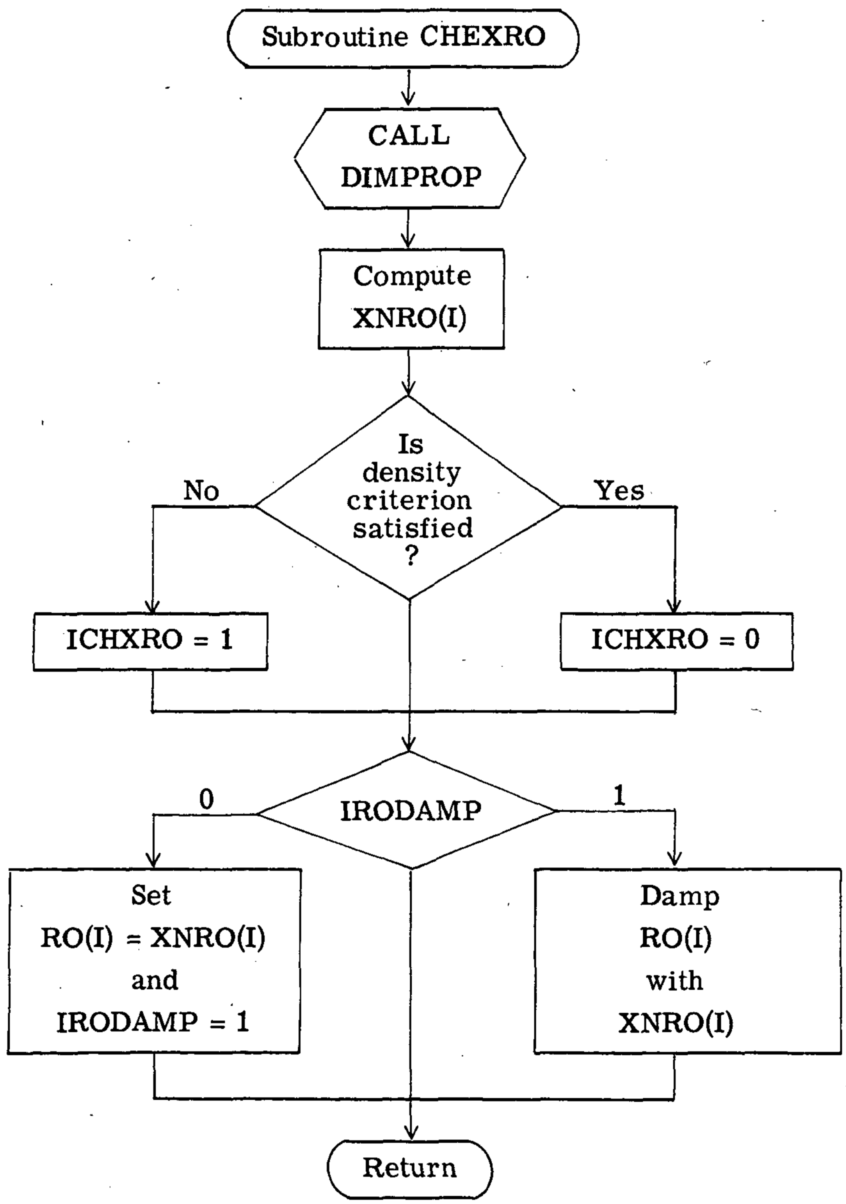
Subroutine FLOW

The FLOW subroutine iterates the fluid-flow equations by computing the $a(\eta)$, $v(\eta)$, $h(\eta)$, $p(\eta)$, and $\bar{a}_F(\eta)$ profiles and shock-wave standoff distance for a given $p(\eta)$ profile. The flow chart for FLOW is given as follows:



Subroutine CHEXRO

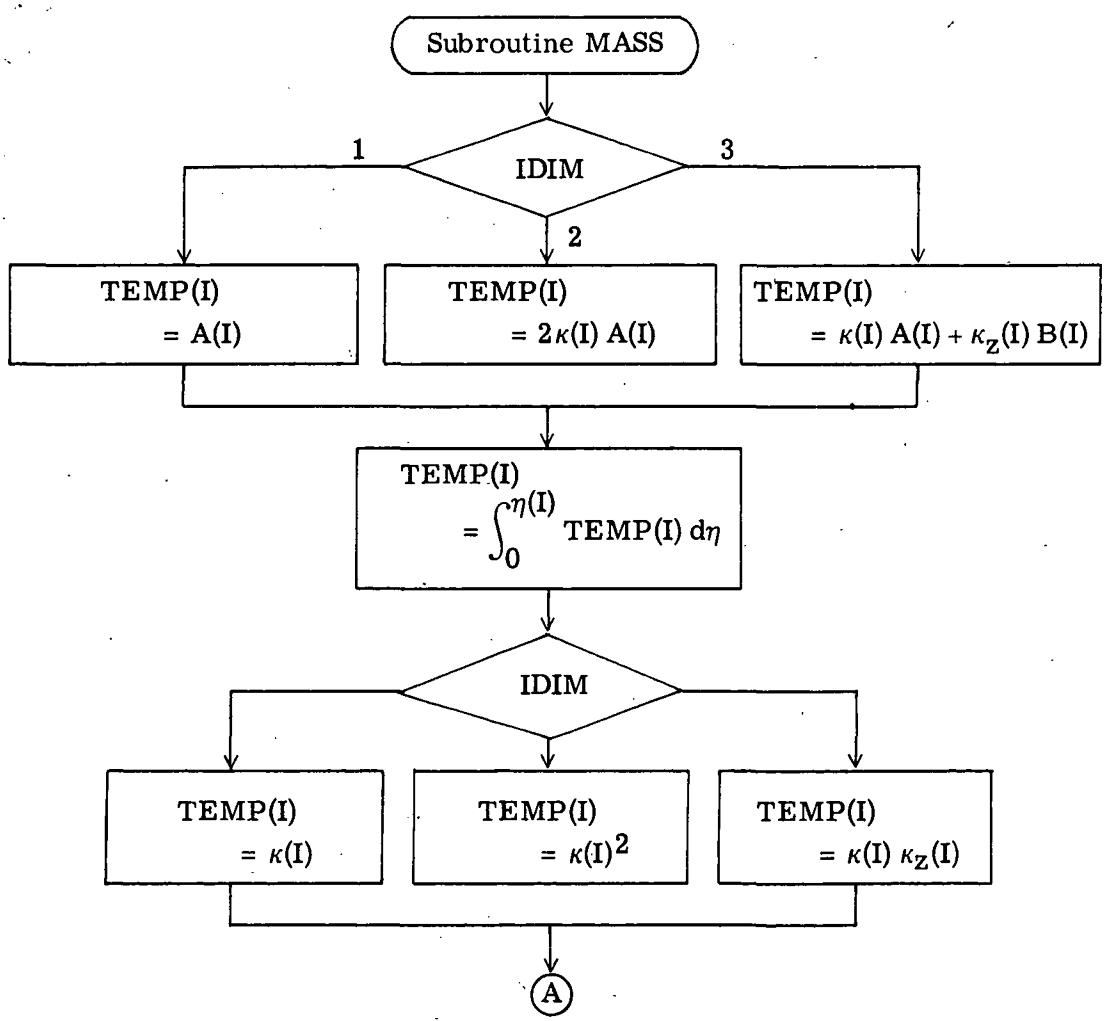
The CHEXRO subroutine computes a density profile RO(I) from the most recent p(I), h(I), and elemental profiles. This RO(I) profile is compared with the previous profile and if the accuracy criterion is satisfied, ICHXRO is set equal to 0; otherwise, ICHXRO is set equal to 1. The RO(I) profile is then damped by using the previous value. Subroutine ICHXRO is used by LEE to determine whether the solution has converged or should be continued. The flow chart for CHEXRO is given as follows:

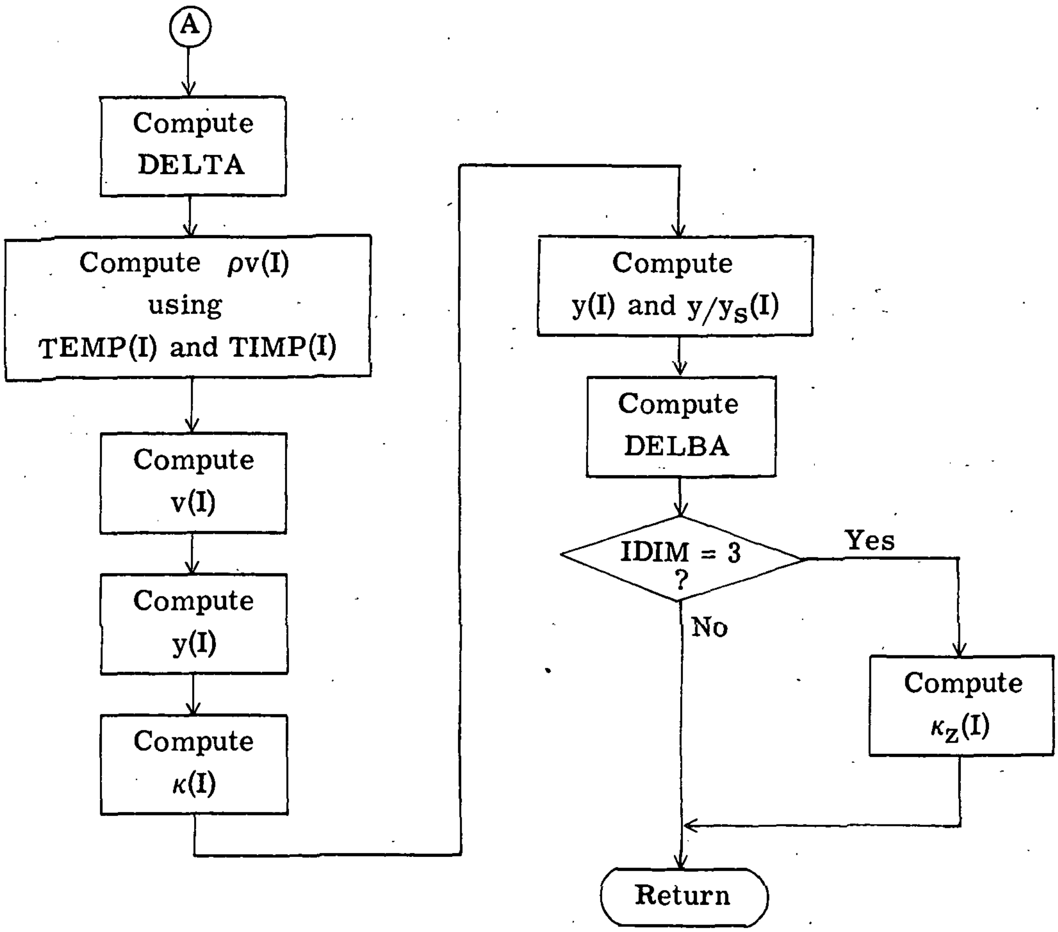


Subroutine MASS

The MASS subroutine integrates the tangential flow gradient $\rho a(\eta)$ to compute the normal mass flow $\rho v\,(\eta)$ in accordance with the continuity equation. Also, the shock-layer thickness and the transformation from η to y are computed. Finally, the physical thickness of the shock layer DELBA is computed. There are three geometry options: IDIM = 1 for two-dimensional flow, IDIM = 2 for axisymmetric flow, and IDIM = 3 for the stagnation streamline of a three-dimensional body.

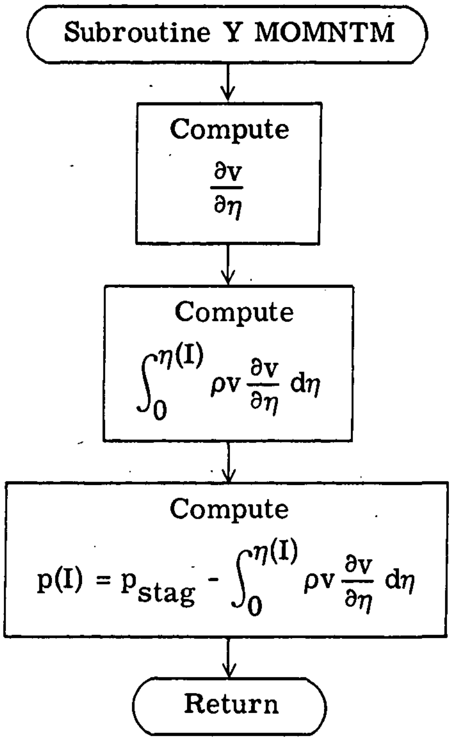
When the computation enters the MASS subroutine, the geometry option flag IDIM is checked, in effect selecting which of equations 9(a), (b), or (c) is to be used. The right-hand side of the selected equation is computed and integrated, without the factor of $\hat{\delta}$. The appropriate scale factors are then divided out, and $\hat{\delta}$ is computed so that $(\rho v)_s = 1$. The remaining computations follow in sequence. The flow chart for MASS is given as follows:




A
Compute DELTA
Compute ρv(I) using TEMP(I) and TIMP(I)
Compute v(I)
Compute y(I)
Compute κ(I)
Compute y(I) and y/y_S(I)
Compute DELBA
IDIM = 3 ?
Yes
No
Compute κ_Z(I)
Return

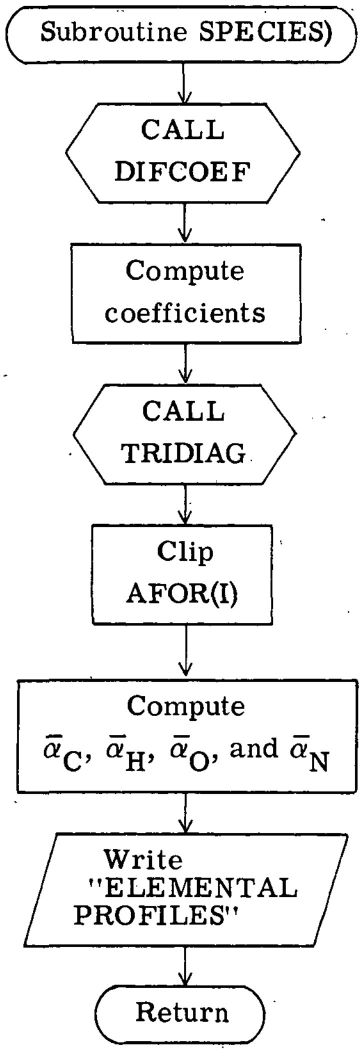
Subroutine Y MOMNTM

The Y MOMNTM subroutine integrates the y-momentum equation by using a density p(I) and velocity profile v(I). The flow chart for Y MOMNTM is given as follows:



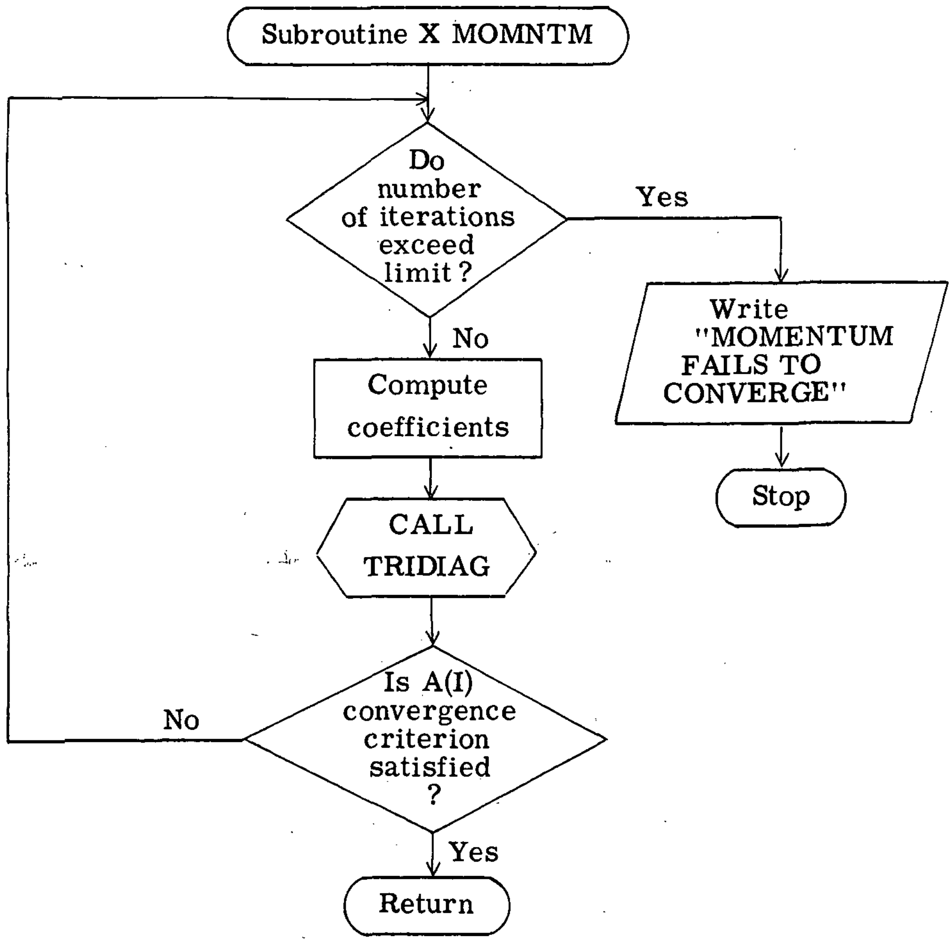
Subroutine SPECIES

The SPECIES subroutine solves the binary diffusion equation for the mass fraction profile $\bar{a}_F(I)$ of the injected mass. The elemental mass fractions $\bar{a}_i(I)$ are then computed from $\bar{a}_F(I)$, where i = C, H, N, and O. The flow chart for SPECIES is given as follows:



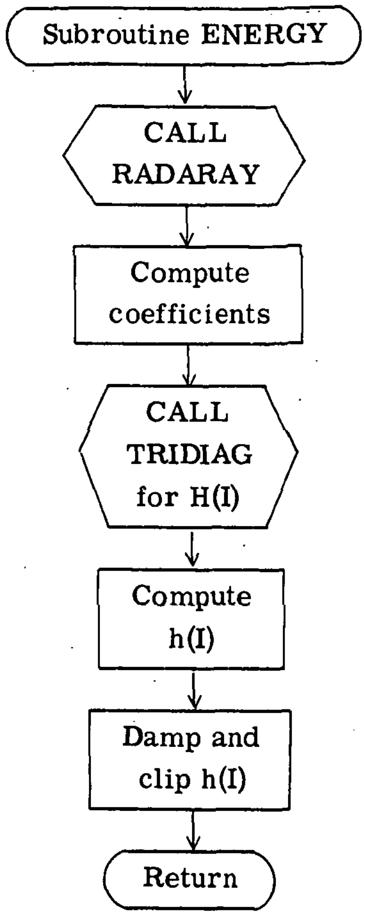
Subroutine X MOMNTM

The X MOMNTM subroutine solves the viscous x-momentum equation for the tangential velocity gradient profile a(I). Because this equation is nonlinear in a(I), a quasilinear approach is used, whereby the solution is computed by iteration. The flow chart for X MOMNTM is given as follows:



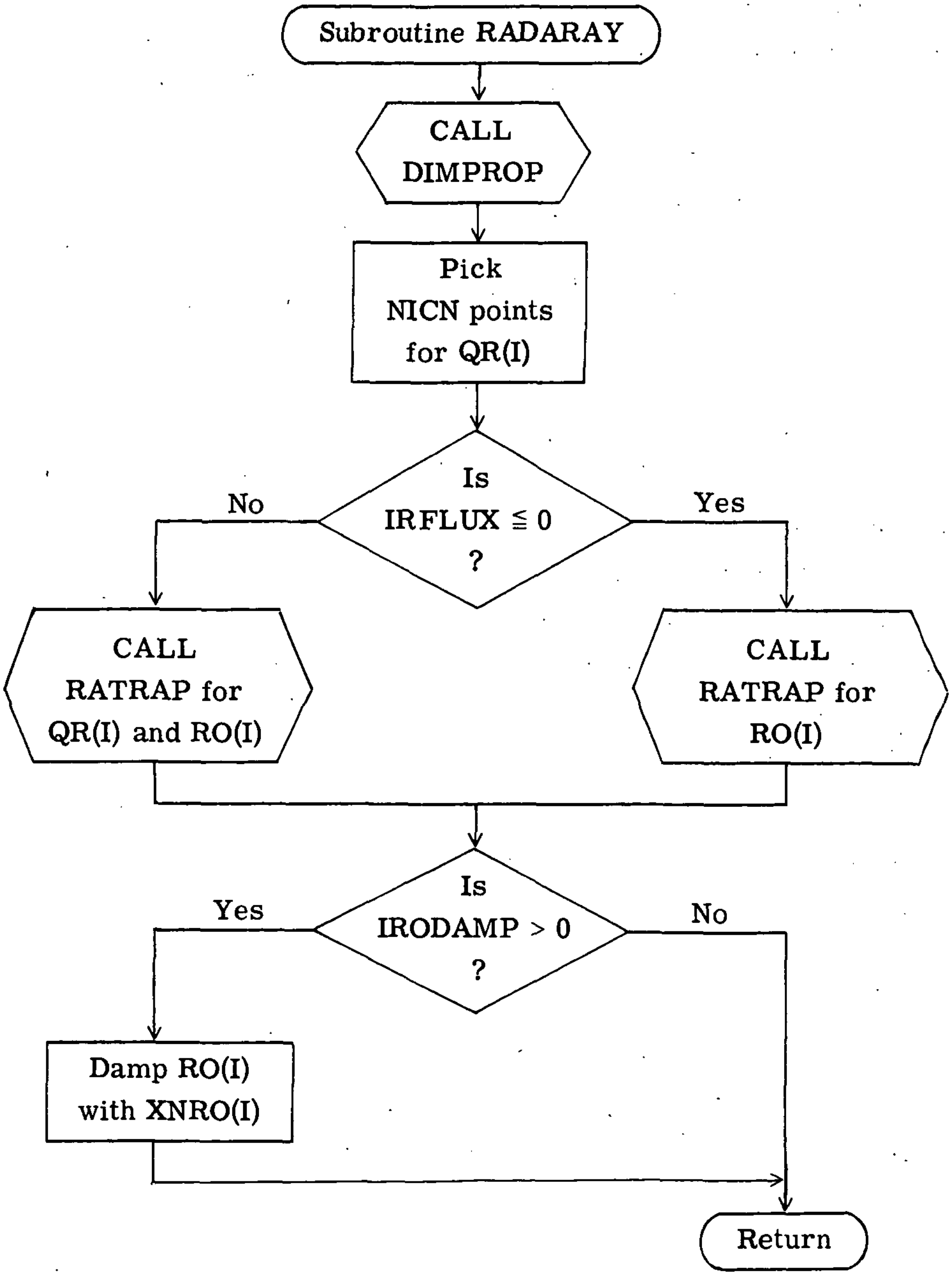
Subroutine ENERGY

The ENERGY subroutine solves the energy equation. The RATRAP code is used to compute radiation heat fluxes $q_{R,y}(I)$, from which $\frac{\partial q_{R,y}}{\partial \eta}$ is computed. The energy equation is solved for H(I), after which h(I) is computed. The flow chart for ENERGY is given as follows:



Subroutine RADARAY

The RADARAY subroutine calls RATRAP for the radiation heat fluxes QR(I) at NIC points and linearly interpolates for the values of QR(I) (where QR(I) = $q_{R,y}(I)$) at the remaining points if IRFLUX > 0. If IRFLUX $\leqq$ 0, QR(I) is set equal to zero. Also, density RO(I) is updated according to the most recent p(I), h(I), and elemental profiles. If damping of the density profile RO(I) is desired, RODAMP is set equal to some positive integer in the NAMELIST NAM4 input. If so, the old density profile, stored in XNRO(I), is used. The flow chart for RADARAY is given as follows:



GAS-PROPERTIES SUBPROGRAMS

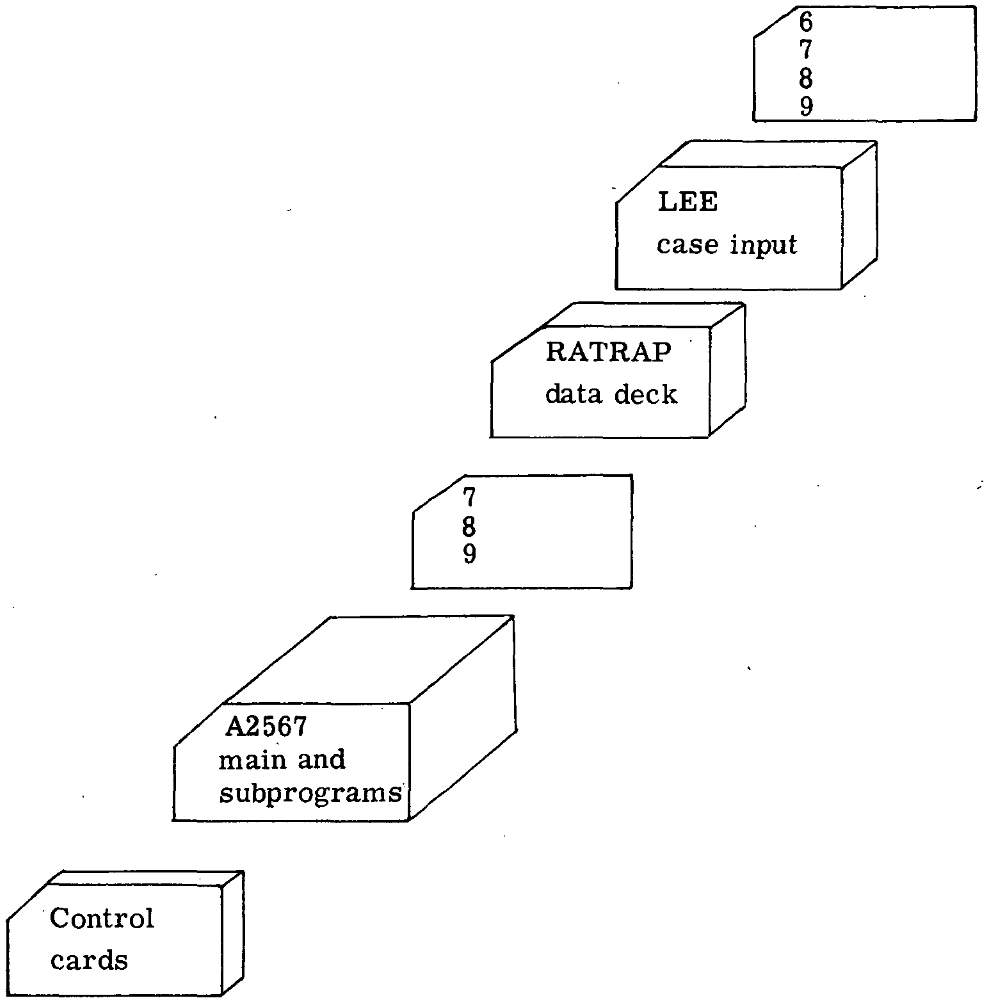
Several of the gas-properties subprograms are for specific gases, for example, air. For other gases, these subprograms should be replaced. The VISCOS and PRANDTL function subprograms select values by using tables from reference 4, which constitute most of the BLOCK DATA subprogram. The DIFCOEF subroutine computes a binary coefficient for a mixture of atomic hydrogen and atomic nitrogen.

The RATRAP subroutine is used to compute the equilibrium composition, density, and temperature of a gaseous mixture of hydrogen, oxygen, nitrogen, and carbon by use of a free-energy minimization subroutine (FEMP).

USAGE

Program Information and Deck Configuration

Program LEE was written in the FORTRAN IV language for the Control Data 6600 computer system under the SCOPE 3.0 operating system. A field length of $150\,000_8$ locations is required. Computing time depends on the blowing rate and the accuracy of the initial profiles assumed; but it is of the order of 20 to 30 minutes for cases with radiation for 21 node points and radiation computed at 11 node points. The vast majority of the time is spent in the very complicated radiation computations. The flow-computation time for fixed density and no radiation is approximately 1 second for each iteration within the FLOW subroutine, so that 2 to 4 seconds are required for a converged solution. The computation time using FEMP for computing the equilibrium chemistry of the air-ablation product mixture and without radiation computations was found to be $1\frac{1}{2}$, $2\frac{1}{2}$, and $3\frac{1}{2}$ minutes for nondimensional blowing rates of 0, -0.1, and -0.2, respectively. In this case, most of the time is spent in chemistry computations. The following sketch shows the deck configuration needed for execution:



Input Description

The RATRAP data deck is supplied with a permanent code and is not normally modified. A listing of the input cards is included with the sample case. Following the RATRAP data deck is the LEE case-input data, which is loaded by NAMELIST. The first group of data, in NAM4, are numbers which may be varied at will but are usually left unchanged. The next group of data, in NAM5, specifies the case of interest. The final data are optional; these are the profiles which may be read in, and they are in NAM1, NAM2, and NAM3. The input symbols are nondimensional except where indicated.

The NAM4 data are given as follows:

N	number of points in flow-field solution, up to 100
IDIM dimension option:	(1) Plane flow (2) Axisymmetric flow (3) Three-dimensional flow
IRODAMP	option for damping density: 0 no damping used on density 1 damping used on density
HDAMP	damping used on static enthalpy profile
RODAMP	damping used on density profile
EP	accuracy criterion for pressure
EROV	accuracy criterion for ρv
EA	accuracy criterion for tangential velocity gradient, $a = \frac{\partial u}{\partial x}$; also used for $b = \frac{\partial w}{\partial z}$
EPSX	accuracy criterion used in solving x-momentum equation
ERO	relative accuracy criterion for density
MAXTIME	maximum number of iterations allowed within FLOW
NIC	number of points for which RATRAP computes radiative fluxes, up to 100
CURV	curvature of body at x = 0 in plane in which x is measured
CURVZ	curvature of body for three-dimensional case at x = 0, z = 0 in plane in which z is measured (used only for IDIM = 3)
BS	$\left(\frac{\partial w}{\partial z}\right)_s$ (used only for IDIM = 3)
D2PDZ	$\left(\frac{\partial^2 p}{\partial z^2}\right)_{x=0,\, z=0}$ (used only for IDIM = 3)

The NAM5 data are given as follows:

IRFLUX	radiation option: 0 no radiation computed 1 radiation computed

IUPDATE	option for inputing initial guess profiles: 0 no profile read in, assume linear 1 read in profile
ALPINJC	mass fraction of carbon in ablation products injected
ALPINJO	mass fraction of oxygen in ablation products injected
ALPINJH	mass fraction of hydrogen in ablation products injected
ALPINJN	mass fraction of nitrogen in ablation products injected
D2DP2C	$\frac{\partial^2 p}{\partial x^2}$ coefficient
AS	value of a at shock wave
PINP	free-stream pressure, dynes/cm^2 (where 1 dyne = 10^{-5} newton)
RHOINP	free-stream density, g/cm^3
UINP	free-stream velocity, cm/sec
RB	body radius at x = 0, cm
ROVW	mass injection rate (negative)
HW	wall static enthalpy
ALPINFN	free-stream mass fraction of nitrogen
ALPINFO	free-stream mass fraction of oxygen
ALPINFC	free-stream mass fraction of carbon
ALPINFH	free-stream mass fraction of hydrogen

The NAM1 data are given as follows:

RO(I)	density profile, initial guess
ROV(I)	profile of mass flow normal to body, initial guess

The NAM2 data are given as follows:

A(I)	profile of tangential velocity gradient, initial guess
H(I)	static enthalpy profile, initial guess

P(I)	pressure profile, initial guess
DELTA	initial guess of transformed shock-layer thickness

The NAM3 data are given as follows:

AFOR(I)	profile of mass fraction of injected species where $I = 1, 2, \ldots N$

A sample set of input data is given in appendix B. This case corresponds to a 304.8-cm (10-ft) nose radius body at an altitude of 60.96 km (200 000 ft) with a velocity of 15.25 km/sec (50 000 ft/sec). This is typical of a manned reentry for a Mars return. Air is being injected at a high rate (0.1 of free-stream value of mass flux).

Discussion of Output

The output of program LEE consists of printing only. Output for the sample case is shown in appendix C. The first page of output, printed by INITIAL, gives the conditions of the case, followed by the postshock conditions. The NAMELIST inputs NAM4 and NAM5 are printed on following pages, also by INITIAL. Next, equilibrium composition profiles as computed by FEMP from the initial pressure and static enthalpy profiles are written by RATRAP. If the radiation option is used, radiative heat fluxes are written. The next page shows the initial flow profiles, printed by INITIAL. For each pass through FLOW, the profiles of the various flow parameters are written. These are followed by profiles of elemental distribution, printed by SPECIES. If the radiation option is used, radiative heat fluxes are printed by RATRAP. Once the iteration for the full-coupled solution has converged, the message "VARIABLE DENSITY SOLN CONVERGED" is printed by LEE. The output symbols are nondimensional, except where indicated.

The symbols for postshock conditions are given as follows:

PSHOCK	static pressure
ROSHOCK	density
VSHOCK	velocity
HSHOCK	static enthalpy

The symbols for flow profiles are given as follows:

ETA	η
Y	y
QR	radiative heat flux
CAPH	total enthalpy
V	v
EMU	μ
PRAN	N_{Pr}
YOYS	y/y_s
ROV	ρv
A	a
P	p, pressure
RO	ρ, density (latest value)
XNRO	ρ, density (previous value)
H	h, static enthalpy (latest value)
XNH	h, static enthalpy (previous value)
T	T', temperature, kelvin
AMFC	$\bar{a}_C$, elemental mass fraction of carbon
AMFN	$\bar{a}_N$, elemental mass fraction of nitrogen
AMFO	$\bar{a}_O$, elemental mass fraction of oxygen
AMFH	$\bar{a}_H$, elemental mass fraction of hydrogen
AFOR	$\bar{a}_F$, total mass fraction of injected products
AONC	total mass fraction of free-stream gas

The symbols for radiative output are given as follows:

TOTAL QMINUS	total radiative heat flux toward shock wave due to atomic emission, W/cm^2
TOTAL QPLUS	total radiative heat flux toward body due to atomic line emission, W/cm^2
QRAD	net radiative heat flux, W/cm^2

Error Messages

The following table gives the error diagnostics as the message and the corresponding subprogram which prints the message:

Message	Diagnosis	Subprogram
FLOW SOLUTION NOT CONVERGED	Fatal	FLOW
MOMENTUM FAILS TO CONVERGE	Fatal	X MOMNTM
FEMP BLEW	Fatal	RATRAP
MAXIMUM ITERATIONS EXCEEDED IN RANHUG	Fatal	RANHUG
ZERO DERIVATIVE IN RANHUG	Fatal	RANHUG
H(I) LESS THAN H(1)	Nonfatal	ENERGY
AFORI(I) LESS THAN 1.E-06	Nonfatal	SPECIES
AFOR(I) GREATER THAN 0.999999	Nonfatal	SPECIES
TEMPERATURE TOO LARGE, T' = ?	Nonfatal	VISCOS
TEMPERATURE TOO LARGE, T' = ?	Nonfatal	PRANDTL
T DID NOT CONVERGE, T' = ?	Nonfatal	FEMP
N-R DID NOT CONVERGE	Nonfatal	FEMP

If the flow solution does not converge, the enthalpy and density profiles should be checked for a sequence of iterations. Adjustment of RODAMP and HDAMP may be indicated. Also, an improved approximation for the initial profiles reduces the number of iterations required. The momentum solution is very reliable. If it should fail to converge, an improved initial profile should be an adequate fix.

Experience with FEMP has shown that the message FEMP BLEW is usually due to negative mass fractions or unrealistically low enthalpies. These problems are alleviated in the computations by the CLIP operation.

No difficulties have been experienced with the computation of postshock conditions. However, diagnostic messages are included in the RANHUG subroutine to provide for this contingency. For some conditions, the Newton-Raphson (N-R) procedure in FEMP has difficulty converging. In this case, the message "N-R DID NOT CONVERGE" is printed and the computation is reinitiated.

The nonfatal error messages are to alert the user to items of which he should be aware, but which do not disrupt the computation.

Langley Research Center,
National Aeronautics and Space Administration,
Hampton, Va., August 7, 1973.

APPENDIX A

PROGRAM LISTING

```
      PROGRAM LEE(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
C
C          LAYER WITH
C          EQUILIBRIUM CHEMISTRY AND
C          EQUILIBRIUM RADIATION
C
      COMMON /CA/ A(101),B(101)
      COMMON /CY/ Y(101),YOYS(101)
      COMMON /CRO/ RO(101)
      COMMON /CETA/ ETA(101)
      COMMON /CV/ V(101)
      COMMON /CP/ P(101)
      COMMON /CROV/ ROV(101)
      COMMON /CEMU/ EMU(101)
      COMMON /CCAPH/ CAPH(101)
      COMMON /CH/ H(101)
      COMMON /CPRAN/ PRAN(101)
      COMMON /CXRO/ XNRO(101)
      COMMON /CXH/ XNH(101)
      COMMON /CQR/ QR(101)
      COMMON /CT/ T(101)
      COMMON /CAMFI/ AMFC(100),AMFN(100),AMFO(100),AMFH(100)
      COMMON /CFORONC/ AFOR(201),AONC(201)
      COMMON /MISCL/ N,IDIM,CURV,DETA,DELTA,PSHOCK,REY,HW,EP,EROV,EA,MAX
     1TIME,MLDIFPT,NIC,ERO
      COMMON /DAMP/ IRODAMP,RODAMP,HDAMP
      COMMON /RADCOMP/ DELBA,IRFLUX
      COMMON /REFVALU/ EMUSHOK
      CALL GETREDY
      CALL INITIAL
1     CALL FLOW
      CALL CHEXRO (ICHXRO)
      IF (ICHXRO.EQ.1) GO TO 1
      WRITE (6,5)
      WRITE (6,6) DELTA,DELBA,REY
      WRITE (6,7)
      DO 2 I=1,N
      WRITE (6,8) ETA(I),Y(I),QR(I),CAPH(I),V(I),EMU(I),PRAN(I),YOYS(I)
2     CONTINUE
      WRITE (6,9)
      DO 3 I=1,N
      WRITE (6,10) ETA(I),ROV(I),A(I),P(I),RO(I),XNRO(I),H(I),XNH(I)
3     CONTINUE
      WRITE (6,11)
      DO 4 I=1,N
      WRITE (6,10) ETA(I),T(I),AMFC(I),AMFN(I),AMFO(I),AMFH(I),AFOR(I),A
     1ONC(I)
4     CONTINUE
      CALL PRINTIT
      STOP
C
C
5     FORMAT (1H0,5X,31HVARIABLE DENSITY SOLN CONVERGED)
6     FORMAT (1H1,5X,6HDELTA=E16.8,2X,6HDELBA=E16.8,2X,4HREY=E16.8)
7     FORMAT (4X,6HETA(I),9X,4HY(I),10X,5HQR(I),8X,7HCAPH(I),10X,4HV(I),
     19X,6HEMU(I),8X,7HPRAN(I),8X,7HYOYS(I))
8     FORMAT (8E14.6)
9     FORMAT (4X,6HETA(I),8X,6HROV(I),9X,4HA(I),10X,4HP(I),10X,5HRO(I),8
     1X,7HXNRO(I),9X,4HH(I),9X,6HXNH(I))
10    FORMAT (8E14.6)
11    FORMAT (4X,6HETA(I),9X,4HT(I),9X,7HAMFC(I),8X,7HAMFN(I),8X,7HAMFO(
     1I),6X,7HAMFH(I),7X,7HAFOR(I),7X,7HAONC(I))
      END
```

```
      SUBROUTINE INITIAL
      COMMON /CA/ A(101),B(101)
      COMMON /CBETA/ BETA(101)
      COMMON /CTEMP/ TEMP(101),TIMP(101)
      COMMON /CY/ Y(101),YOYS(101)
      COMMON /CETA/ ETA(101)
      COMMON /CP/ P(101)
      COMMON /CRO/ RO(101)
      COMMON /CROV/ ROV(101)
      COMMON /CSCH/ SCH(101),SCI(101)
      COMMON /CV/ V(101)
      COMMON /CH/ H(101)
      COMMON /CT/ T(101)
      COMMON /CFORONC/ AFOR(201),AONC(201)
      COMMON /FRSTRM/ HIN,PIN,RHOINP,UINP,RB
      COMMON /CAMFI/ AMFC(100),AMFN(100),AMFO(100),AMFH(100)
      COMMON /MFINF/ ALPINFC,ALPINFN,ALPINFO,ALPINFH
      COMMON /MFINJ/ ALPINJC,ALPINJN,ALPINJO,ALPINJH
      COMMON /RANK/ ROSHOCK,VSHOCK,HSHOCK,TSHOCK
      COMMON /MISCL/ N,IDIM,CURV,DETA,DELTA,PSHOCK,REY,HW,EP,EROV,EA,MAX
     1TIME,MLDIFPT,NIC,ERO
      COMMON /REFVALU/ EMUSHOK
      COMMON /RADCOMP/ DELBA,IRFLUX
      COMMON /DAMP/ IRODAMP,RODAMP,HDAMP
      COMMON /CD2DP2/ D2DP2C
      COMMON /C3D/ CURVZ,BS,BETAZ(101)
      NAMELIST /NAM4/ N,IDIM,IRODAMP,HDAMP,RODAMP,EP,EROV,EA,EPSX,ERO,MA
     1XTIME,NIC,CURV,CURVZ,BS,D2PDZ
      NAMELIST /NAM5/ IRFLUX,IUPDATE,ALPINJC,ALPINJO,ALPINJH,ALPINJN,D2D
     1P2C,AS,PINP,RHOINP,UINP,RB,ROVW,HW,ALPINFN,ALPINFO,ALPINFC,ALPINFH
      READ (5,NAM4) $READ(5,NAM5)
      WRITE (6,20)
      IF (IRFLUX.GT.0) WRITE (6,19)
      WRITE (6,17) RB,UINP,RHOINP,AS,D2DP2C,PINP,ROVW,ALPINJH,ALPINJC,AL
     1PINJN,ALPINJO
      WRITE (6,18) ALPINFN,ALPINFO,ALPINFC,ALPINFH
      PIN=PINP/(RHOINP*UINP*UINP)
      HIN=3.5*PIN+0.5
      CALL RANHUG
      WRITE (6,NAM4) $WRITE(6,NAM5)
      REY=RHOINP*UINP*RB/EMUSHCK
      RO(N)=RCSHOCK
      XN=N-1
      DETA=1./XN
      ROV(N)=1.
      A(1)=0.
      TEMP=AS-A(1)
      ETA(1)=0.
      DO 1 I=2,N
    1 ETA(I)=ETA(I-1)+DETA
      DO 2 I=1,N
    2 BETA(I)=D2DP2(ETA(I))
      IF (IDIM.NE.3) GO TO 4
      DO 3 I=1,N
      BETAZ(I)=D2PDZ
    3 B(I)=BS*ETA(I)
    4 CONTINUE
      IF (IUPDATE.GT.0) GO TO 14
      DELTA=1.
      DELBA=.05*RB
      DO 5 I=1,N
      ROV(I)=(ROV(N)-ROV(1))*ETA(I)+ROV(1)
      ROV(I)=ROV(1)+(1.-ROV(1))*ETA(I)*ETA(I)
      H(I)=(HSHOCK-HW)*ETA(I)+HW
      A(I)=A(1)+TEMP*ETA(I)
    5 CONTINUE
      DO 6 I=2,N
```

```
6     RO(I)=ROSHOCK/ETA(I)
      RO(1)=RO(2)+5.
      DO 7 I=1,N
      FOREIGN=0.
      IF (ROV(I).GT..0) FOREIGN=1.
      AFOR(I)=1.-FOREIGN
7     AONC(I)=FOREIGN
      DO 8 I=1,N
      AMFC(I)=AFOR(I)*ALPINJC+AONC(I)*ALPINFC
      AMFN(I)=AFOR(I)*ALPINJN+AONC(I)*ALPINFN
      AMFO(I)=AFOR(I)*ALPINJO+AONC(I)*ALPINFO
      AMFH(I)=AFOR(I)*ALPINJH+AONC(I)*ALPINFH
8     CONTINUE
      DO 9 I=1,N
      V(I)=ROV(I)/RO(I)
      SCH(I)=1.+DELTA*ETA(I)/RO(I)
      SCI(I)=1.+DELTA*ETA(I)/RO(I)
9     CONTINUE
      TEMP=-.5*V(N)
      PSTAG=PSHOCK-TEMP
      DO 10 I=1,N
10    P(I)=PSTAG+TEMP*ETA(I)**2
      DO 11 I=1,N
11    TEMP(I)=1./RO(I)
      CALL TINT (N,DETA,TEMP,TIMP)
      DO 12 I=1,N
      Y(I)=DELTA*TIMP(I)
      SCH(I)=1.+CURV*DELTA*TIMP(I)
12    CONTINUE
      DO 13 I=1,N
      YOYS(I)=Y(I)/Y(N)
13    CONTINUE
      DELBA=Y(N)*RB
      GO TO 15
14    CONTINUE
      CALL UPDATE
15    CONTINUE
      WRITE (6,21)
      CALL PRINTIT
      WRITE (6,22)
      WRITE (6,23) DELTA,DELBA,REY,EMUSHOK
      WRITE (6,24)
      DO 16 I=1,N
      WRITE (6,25) ETA(I),ROV(I),A(I),P(I),RO(I),H(I),T(I),AFOR(I)
16    CONTINUE
      RETURN
C
C
17    FORMAT (//8X,12HBODY RADIUS=,E13.5,25H CM, FREESTREAM VELOCITY=,E1
     13.5,17H CM/SEC, DENSITY=,E13.5,5H G/CC/9X,18HA SHOCK (NON DIM)=,F8
     2.5,19H,    BETA (NON DIM)=,F8.5,1H,,16X,9HPRESSURE=,E13.5,12H DYNES
     3/SQ CM/9X,24HABLATION RATE (NON DIM)=,F8.5,30H,    MASS FRACTIONS-
     4HYDROGEN =,F7.5,10H,  OXYGEN=,F7.5,13H,  NITROGEN =,F7.5,11H,  CAR
     5BON =,F7.5)
18    FORMAT (9X,28HFREESTREAM COMPOSITION- N=   ,F7.5,7H,  O=  ,F7.5,7H,
     1 C=   ,F7.5,7H,  H=  ,F7.5)
19    FORMAT (50X,14HWITH RADIATION)
20    FORMAT (1H1,///38X,38HVISCOUS STAGNATION STREAMLINE SOLUTION/)
21    FORMAT (1H1,////11X,40HINITIAL EQUILIBRIUM COMPOSITION PROFILES/)
22    FORMAT (1H1,////11X,21HINITIAL FLOW PROFILES)
23    FORMAT (//5X,6HDELTA=E16.8,2X,6HDELBA=E16.8,2X,4HREY=E16.8,2X,21H
     1 VISCOSITY(SHOCK)=   ,E16.8)
24    FORMAT (4X,6HETA(I),8X,6HROV(I),9X,4HA(I),10X,4HP(I),10X,5HRO(I),8
     1X,4HH(I),8X,4HT(I)8X,7HAFOR(I))
25    FORMAT (8E14.6)
      END
```

```
      SUBROUTINE RANHUG
      COMMON /CAMFI/ AMFC(100),AMFN(100),AMFO(100),AMFH(100)
      COMMON /RANK/ ROSHOCK,VSHOCK,HSHOCK,TSHOCK
      COMMON /MISCL/ N,IDIM,CURV,DETA,DELTA,PSHOCK,REY,HW,EP,EROV,EA,MAX
     1TIME,MLDIFPT,NIC,ERO
      COMMON /FRSTRM/ HIN,PIN,RHOINP,UINP,RB
      COMMON /MFINF/ ALPINFC,ALPINFN,ALPINFO,ALPINFH
      COMMON /MFINJ/ ALPINJC,ALPINJN,ALPINJC,ALPINJH
      COMMON /REFVALU/ EMUSHOK
      EXTERNAL FOFXRO
      EXTERNAL VISCOS
      PSHOCK=0.95
      VSHOCK=1.+PIN-PSHOCK
      HSHOCK=HIN-(VSHOCK*VSHOCK/2.)
      CALL DIMPROP (PSHOCK,HSHOCK,PSHOKP,HSHOKP)
      AINFC=ALPINFC
      AINFN=ALPINFN
      AINFO=ALPINFO
      TSHOKP=14000.
      TSHOCK=TSHOKP
      ROSHOCK=RHOSY(PSHOKP,HSHOKP*3.483E-07,AINFC,AINFN,AINFO,TSHOKP)
      CALL ITR1 (ROSHOCK,.1,FOFXRO,.001,.001,20,ICODE)
      IF (ICODE.EQ.0) GO TO 1
      IF (ICODE.EQ.1) WRITE (6,2)
      IF (ICODE.EQ.2) WRITE (6,3)
      STOP
1     WRITE (6,4) PSHOCK,ROSHOCK,VSHOCK,HSHOCK
C
      EMUSHOK=VISCOS(TSHOKP,PSHOKP)
      RETURN
C
C
2     FORMAT (5X,37HMAXIMUM ITERATIONS EXCEEDED IN RANHUG)
3     FORMAT (5X,25HZERO DERIVATIVE IN RANHUG)
4     FORMAT (////5X,15HRANHUG SOLUTION//12H     PSHOCK=E16.8,2X,8HROSHO
     1CK=E16.8,2X,7HVSHOCK=E16.8,2X,7HHSHOCK=E16.8)
      END
```

```
      FUNCTION FOFXRO (ROSHOC)
      COMMON /RANK/ ROSHOCK,VSHOCK,HSHOCK,TSHOCK
      COMMON /FRSTRM/ HIN,PIN,RHOINP,UINP,RB
      COMMON /MFINF/ ALPINFC,ALPINFN,ALPINFO,ALPINFH
      COMMON /MISCL/ N,IDIM,CURV,DETA,DELTA,PSHOCK,REY,HW,EP,EROV,EA,MAX
     1TIME,MLDIFPT,NIC,ERO
      VSHOCK=1./ROSHOC
      PSHOCK=PIN+1.-VSHOCK
      HSHOCK=HIN-(VSHOCK*VSHOCK/2.)
      CALL DIMPROP (PSHOCK,HSHOCK,PSHOKP,HSHOKP)
      AINFC=ALPINFC
      AINFN=ALPINFN
      AINFO=ALPINFO
      TSHOKP=TSHOCK
      FOFXRO=RHOSY(PSHOKP,HSHOKP*3.483E-07,AINFC,AINFN,AINFO,TSHOKP)
      TSHOCK=TSHOKP
      RETURN
      END
```

```
      SUBROUTINE UPCATE
      COMMON /CA/ A(101),B(101)
      COMMON /CY/ Y(101),YOYS(101)
      COMMON /CETA/ ETA(101)
      COMMON /CRO/ RO(101)
      COMMON /CV/ V(101)
      COMMON /CP/ P(101)
      COMMON /CROV/ ROV(101)
      COMMON /CH/ H(101)
      COMMON /CSCH/ SCH(101),SCI(101)
      COMMON /CTEMP/ TEMP(101),TIMP(101)
      COMMON /CT/ T(101)
      COMMON /CAMFI/ AMFC(100),AMFN(100),AMFO(100),AMFH(100)
      COMMON /CFORONC/ AFOR(201),AONC(201)
      COMMON /FRSTRM/ HIN,PIN,RHOINP,UINP,RB
      COMMON /MFINF/ ALPINFC,ALPINFN,ALPINFO,ALPINFH
      COMMON /MFINJ/ ALPINJC,ALPINJN,ALPINJO,ALPINJH
      COMMON /MISCL/ N,IDIM,CURV,DETA,DELTA,PSHOCK,REY,HW,EP,EROV,EA,MAX
     1TIME,MLDIFPT,NIC,ERO
      COMMON /RADCOMP/ DELBA,IRFLUX
      NAMELIST /NAM1/ RO,ROV
      NAMELIST /NAM2/ A,H,P,DELTA
      NAMELIST /NAM3/ AFOR
      READ (5,NAM1)
      READ (5,NAM2)
      READ (5,NAM3)
      DO 1 I=1,N
    1 TEMP(I)=1./RO(I)
      CALL TINT (N,DETA,TEMP,TIMP)
      DO 2 I=1,N
      Y(I)=DELTA*TIMP(I)
      SCH(I)=1.+CURV*DELTA*TIMP(I)
      SCI(I)=SCH(I)
    2 CONTINUE
      DO 3 I=1,N
      YOYS(I)=Y(I)/Y(N)
    3 CONTINUE
      DELBA=Y(N)*RB
      DO 4 I=1,N
    4 AONC(I)=1.-AFOR(I)
      DO 5 I=1,N
      AMFC(I)=AFOR(I)*ALPINJC+AONC(I)*ALPINFC
      AMFN(I)=AFOR(I)*ALPINJN+AONC(I)*ALPINFN
      AMFO(I)=AFOR(I)*ALPINJO+AONC(I)*ALPINFO
      AMFH(I)=AFOR(I)*ALPINJH+AONC(I)*ALPINFH
    5 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE FLOW
      COMMON /CA/ A(101),B(101)
      COMMON /CBETA/ BETA(101)
      COMMON /CY/ Y(101),YOYS(101)
      COMMON /CRO/ RO(101)
      COMMON /CETA/ ETA(101)
      COMMON /CV/ V(101)
      COMMON /CP/ P(101)
      COMMON /CROV/ ROV(101)
      COMMON /CDV/ DV(101)
      COMMON /CSCH/ SCH(101),SCI(101)
      COMMON /CTEMP/ TEMP(101),TIMP(101)
      COMMON /CEMU/ EMU(101)
      COMMON /CROMU/ ROMU(101)
      COMMON /CCAPH/ CAPH(101)
      COMMON /CH/ H(101)
      COMMON /CPRAN/ PRAN(101)
      COMMON /CQR/ QR(101)
      COMMON /CXRO/ XNRO(101)
      COMMON /CXH/ XNH(101)
      COMMON /CT/ T(101)
      COMMON /CAMFI/ AMFC(100),AMFN(100),AMFO(100),AMFH(100)
      COMMON /CFORONC/ AFOR(201),AONC(201)
      COMMON /MISCL/ N,IDIM,CURV,DETA,DELTA,PSHOCK,REY,HW,EP,EROV,EA,MAX
     1TIME,MLDIFPT,NIC,ERO
      COMMON /MFINF/ ALPINFC,ALPINFN,ALPINFO,ALPINFH
      COMMON /MFINJ/ ALPINJC,ALPINJN,ALPINJO,ALPINJH
      COMMON /RADCOMP/ DELBA,IRFLUX
      COMMON /DAMP/ IRODAMP,RODAMP,HDAMP
      COMMON /C3D/ CURVZ,BS,BETAZ(101)
      COMMON /REFVALU/ EMUSHOK
      DIMENSION TAMP(101)
      IZMOM=0
      NTIMES=0
1     CONTINUE
      WRITE (6,8)
      WRITE (6,9) DELTA,DELBA,REY
      WRITE (6,10)
      DO 2 I=1,N
      WRITE (6,11) ETA(I),Y(I),QR(I),CAPH(I),V(I),EMU(I),PRAN(I),YOYS(I)
2     CONTINUE
      WRITE (6,12)
      DO 3 I=1,N
      WRITE (6,13) ETA(I),ROV(I),A(I),P(I),RO(I),XNRO(I),H(I),XNH(I)
3     CONTINUE
      WRITE (6,14)
      DO 4 I=1,N
      WRITE (6,13) ETA(I),T(I),AMFC(I),AMFN(I),AMFO(I),AMFH(I),AFOR(I),B
     1(I)
4     CONTINUE
      CALL SET (N,ROV,TAMP)
      CALL MASS
      ICONT=ITEST(N,ROV,TAMP,EROV)
      CALL SET (N,P,TAMP)
      CALL YMOMNTM
      IYMOM=ITEST(N,P,TAMP,EP)
      CALL SPECIES
      CALL VISPRAN
      CALL SET (N,A,TAMP)
      CALL XMOMNTM (A,BETA,CURV)
      IXMOM=ITEST(N,A,TAMP,EA)
      IF (IDIM.NE.3) GO TO 5
      CALL SET (N,B,TAMP)
      CALL XMOMNTM (A,BETAZ,CURVZ)
      IZMOM=ITEST(N,B,TAMP,EA)
5     CONTINUE
      CALL ENERGY
      IF ((ICONT.NE.0).OR.(IYMOM.NE.0).OR.(IXMOM.NE.0).OR.(IZMOM.NE.0))
     1GO TO 6
      GO TO 7
6     NTIMES=NTIMES+1
      IF (NTIMES.LE.MAXTIME) GO TO 1
```

```
      WRITE (6,16)
      STOP
7     CONTINUE
      WRITE (6,15)
      RETURN
C
C
8     FORMAT (1H1,////11X,13HFLOW PROFILES)
9     FORMAT (///5X,6HDELTA=E16.8,2X,6HDELBA=E16.8,2X,4HREY=E16.8/)
10    FORMAT (4X,6HETA(I),9X,4HY(I),10X,5HQR(I),8X,7HCAPH(I),10X,4HV(I),
     19X,6HEMU(I),8X,7HPRAN(I),8X,7HYOYS(I))
11    FORMAT (8E14.6)
12    FORMAT (4X,6HETA(I),8X,6HROV(I),9X,4HA(I),10X,4HP(I),10X,5HRO(I),8
     1X,7HXNRO(I),9X,4HH(I),9X,6HXNH(I))
13    FORMAT (8E14.6)
14    FORMAT (4X,6HETA(I),9X,4HT(I),9X,7HAMFC(I),8X,7HAMFN(I),8X,7HAMFO(
     1I),6X,7HAMFH(I),7X,7HAFOR(I),7X,4HB(I))
15    FORMAT (5X,23HFLOW SOLUTION CONVERGED)
16    FORMAT (5X,27HFLOW SOLUTION NOT CONVERGED)
      END
```

```
      SUBROUTINE CHEXRO (ICHXRO)
      COMMON /CH/ H(101)
      COMMON /CP/ P(101)
      COMMON /CRO/ RO(101)
      COMMON /CXRO/ XNRO(101)
      COMMON /CAMFI/ AMFC(100),AMFN(100),AMFO(100),AMFH(100)
      COMMON /MISCL/ N,IDIM,CURV,DETA,DELTA,PSHOCK,REY,HW,EP,EROV,EA,MAX
     1TIME,MLDIFPT,NIC,ERO
      COMMON /DAMP/ IRODAMP,RODAMP,HDAMP
      DIMENSION ABRAT(101), RATIO(101)
      DIMENSION PP(101), HP(101)
      DIMENSION AMFCP(100), AMFNP(100), AMFOP(100)
      EXTERNAL RHOSY
      ICHXRO=0
      CALL SET (N,AMFC,AMFCP)
      CALL SET (N,AMFN,AMFNP)
      CALL SET (N,AMFO,AMFOP)
      TGUES=4000.
      DO 1 I=1,N
      CALL DIMPROP (P(I),H(I),PP(I),HP(I))
      XNRO(I)=RHOSY(PP(I),HP(I)*3.483E-07,AMFCP(I),AMFNP(I),AMFOP(I),TGU
     1ES)
1     CONTINUE
      DO 2 I=1,N
      RATIO(I)=(XNRO(I)-RO(I))/XNRO(I)
      ABRAT(I)=ABS(RATIO(I))
      IF (ABRAT(I).GT.ERO) ICHXRO=1
2     CONTINUE
      IF (IRODAMP.GT.0) GO TO 4
      DO 3 I=1,N
3     RO(I)=XNRO(I)
      IRODAMP=1
      GO TO 6
4     CONTINUE
      DO 5 I=1,N
      RO(I)=RODAMP*RO(I)+(1.-RODAMP)*XNRO(I)
5     CONTINUE
6     CONTINUE
      RETURN
      END
```

```
      SUBROUTINE MASS
      COMMON /CA/ A(101),B(101)
      COMMON /CETA/ ETA(101)
      COMMON /CY/ Y(101),YOYS(101)
      COMMON /CRO/ RO(101)
      COMMON /CROV/ ROV(101)
      COMMON /CSCH/ SCH(101),SCI(101)
      COMMON /CTEMP/ TEMP(101),TIMP(101)
      COMMON /CV/ V(101)
      COMMON /MISCL/ N,IDIM,CURV,DETA,DELTA,PSHOCK,REY,HW,EP,EROV,EA,MAX
     1TIME,MLDIFPT,NIC,ERO
      COMMON /FRSTRM/ HIN,PIN,RHOINP,UINP,RB
      COMMON /RADCOMP/ DELBA,IRFLUX
      COMMON /C3D/ CURVZ,BS,BETAZ(101)
C
C        MASS CONSERVATION PACKAGE
C     COMPRESSIBLE ETA TRANSFORMATION
C
      IF (IDIM.EQ.2) GO TO 2
      IF (IDIM.EQ.3) GO TO 4
      DO 1 I=1,N
1     TEMP(I)=A(I)
      GO TO 6
2     DO 3 I=1,N
3     TEMP(I)=2.*A(I)*SCH(I)
      GO TO 6
4     DO 5 I=1,N
5     TEMP(I)=A(I)*SCH(I)+B(I)*SCI(I)
6     CONTINUE
      CALL TINT (N,CETA,TEMP,TIMP)
      IF (IDIM.EQ.2) GO TO 7
      IF (IDIM.EQ.3) GO TO 9
      CALL SET (N,SCH,TEMP)
      GO TO 11
7     DO 8 I=1,N
8     TEMP(I)=SCH(I)**2
      GO TO 11
9     DO 10 I=1,N
10    TEMP(I)=SCH(I)*SCI(I)
11    CONTINUE
      DELTA=(TEMP(N)-ROV(1))/TIMP(N)
      DO 12 I=1,N
12    ROV(I)=(DELTA*TIMP(I)+ROV(1))/TEMP(I)
      DO 13 I=1,N
C
C        MASS CONSERVATION SATISFIED
C
13    V(I)=ROV(I)/RC(I)
      DO 14 I=1,N
14    TEMP(I)=1./RO(I)
      CALL TINT (N,DETA,TEMP,TIMP)
      DO 15 I=1,N
      Y(I)=DELTA*TIMP(I)
      SCH(I)=1.+CURV*DELTA*TIMP(I)
15    CONTINUE
      DO 16 I=1,N
      YOYS(I)=Y(I)/Y(N)
16    CONTINUE
      DELBA=Y(N)*RB
      IF (ICIM.EQ.2) GO TO 18
      IF (ICIM.EQ.3) GO TO 20
      DO 17 I=1,N
17    SCI(I)=1.
      GO TO 22
18    DO 19 I=1,N
19    SCI(I)=SCH(I)
      GO TO 22
20    DO 21 I=1,N
21    SCI(I)=1.+CURVZ*DELTA*TIMP(I)
22    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE YMOMNTM
      COMMON /CETA/ ETA(101)
      COMMON /CP/ P(101)
      COMMON /CRO/ RO(101)
      COMMON /CROV/ ROV(101)
      COMMON /CDV/ DV(101)
      COMMON /CTEMP/ TEMP(101),DP(101)
      COMMON /CV/ V(101)
      COMMON /MISCL/ N,IDIM,CURV,DETA,DELTA,PSHOCK,REY,HW,EP,EROV,EA,MAX
     1TIME,MLDIFPT,NIC,ERO
C         THIS SUBROUTINE INTEGRATES THE Y MOMENTUM EQUATION TO GIVE
C           P(Y)
      CALL DIFT (V,ETA,DV,N)
      DO 1 I=1,N
1     TEMP(I)=ROV(I)*DV(I)
      CALL TINT (N,DETA,TEMP,DP)
      PSTAG=PSHOCK+DP(N)
      DO 2 I=1,N
      P(I)=PSTAG-DP(I)
2     CONTINUE
      RETURN
      END
```

```
      SUBRCUTINE SPECIES
      COMMON /CRO/ RO(101)
      COMMON /CT/ T(101)
      COMMON /CETA/ ETA(101)
      COMMON /CROV/ ROV(101)
      COMMON /CSCH/ SCH(101),SCI(101)
      COMMON /CD12/ D12(101)
      COMMON /CAMFI/ AMFC(100),AMFN(100),AMFO(100),AMFH(100)
      COMMON /CFORONC/ AFOR(201),AONC(201)
      COMMON /MISCL/ N,IDIM,CURV,DETA,DELTA,PSHOCK,REY,HW,EP,EROV,EA,MAX
     1TIME,MLDIFPT,NIC,ERO
      COMMON /MFINJ/ ALPINJC,ALPINJN,ALPINJO,ALPINJH
      COMMON /MFINF/ ALPINFC,ALPINFN,ALPINFO,ALPINFH
      DIMENSION AS(101), BS(101), CS(101), CS(101)
      DIMENSION RK(201), D12K(201), RVK(201), SHK(201)
      DIMENSION ETADIF(201), AK(201), BK(201), CK(201), DK(201)
      CIMENSICN DI(201), CO(201)
      CALL CIFCOEF
      AINF=1.0
      AINJ=1.0
      NLI=N-1
      CO 1 I=1,N
      DI(I)=SCH(I)*RO(I)*RO(I)*D12(I)*SCI(I)
1     CO(I)=DETA*SCH(I)*ROV(I)*SCI(I)
C     BOUNDARY CONDITIONS
C     DIFFUSION OF SHOCK LAYER SPECIES TO WALL PERMITTED
      AS(1)=0.
      BS(1)=-DI(1)+CO(1)*DELTA
      CS(1)=DI(1)
      DS(1)=DELTA*CO(1)*AINJ
      AS(N)=0.
      BS(N)=1.0
      CS(N)=0.
      DS(N)=0
C
      DO 2 I=2,NLI
      SIGN=0.
      IF (ROV(I).GT.0.) SIGN=1.0
      AS(I)=DI(I-1)+DI(I)-2.*CC(I)*DELTA*(1.-SIGN)
      BS(I)=-(DI(I-1)+2*DI(I)+DI(I+1))
      BS(I)=BS(I)+2.*CO(I)*DELTA*(1.-SIGN)-2.*CO(I)*DELTA*SIGN
      CS(I)=DI(I)+DI(I+1)+2.*CO(I)*DELTA*SIGN
2     DS(I)=0.
      CALL TRIDIAG (N,AS,BS,CS,DS)
      CO 3 I=1,N
      AFOR(I)=DS(I)
3     AONC(I)=1.-AFOR(I)
      CO 7 I=1,N
      IF (AFOR(I).LT.1.E-06) 4,5
4     WRITE (6,10)
      WRITE (6,11) ETA(I),AFOR(I)
      AFOR(I)=0.
      AONC(I)=1.0
5     CONTINUE
      IF (AFOR(I).GT..999999) 6,7
6     WRITE (6,12)
      WRITE (6,11) ETA(I),AFOR(I)
      AFOR(I)=1.0
      AONC(I)=0.
      WRITE (6,11)
7     CONTINUE
      DO 8 I=1,N
      AMFC(I)=AFOR(I)*ALPINJC+AONC(I)*ALPINFC
      AMFN(I)=AFOR(I)*ALPINJN+AONC(I)*ALPINFN
      AMFO(I)=AFOR(I)*ALPINJO+AONC(I)*ALPINFO
      AMFH(I)=AFOR(I)*ALPINJH+AONC(I)*ALPINFH
      AMFC(I)=AMFC(I)/(AMFC(I)+AMFN(I)+AMFO(I)+AMFH(I))
      AMFN(I)=AMFN(I)/(AMFC(I)+AMFN(I)+AMFO(I)+AMFH(I))
      AMFO(I)=AMFO(I)/(AMFC(I)+AMFN(I)+AMFO(I)+AMFH(I))
8     AMFH(I)=AMFH(I)/(AMFC(I)+AMFN(I)+AMFO(I)+AMFH(I))
      WRITE (6,13)
```

```
      DO 9 I=1,N
9     WRITE (6,14) ETA(I),AMFC(I),AMFN(I),AMFO(I),AMFH(I)
      RETURN
C
C
10    FORMAT (1H0,5X,24HAFOR(I) LESS THAN 1.E-06)
11    FORMAT (2E14.6)
12    FORMAT (1H0,5X,28HAFOR(I) GREATER THAN .999999)
13    FORMAT (//10X,18HELEMENTAL PROFILES/5X,6HETA(I),6X,9HCARBON(I),4X,
     111HNITROGEN(I),3X,9HOXYGEN(I),5X,11HHYDROGEN(I)/)
14    FORMAT (8E14.6)
      END
```

```
      SUBROUTINE XMOMNTM (A,BETA,Q)
      COMMON /CETA/ ETA(101)
      COMMON /CRO/ RO(101)
      COMMON /CROV/ ROV(101)
      COMMON /CDV/ DV(101)
      COMMON /CEMU/ EMU(101)
      COMMON /CROMU/ ROMU(101)
      COMMON /CSCH/ SCH(101),SCI(101)
      COMMON /MISCL/ N,IDIM,CURV,DETA,DELTA,PSHOCK,REY,HW,EP,EROV,EA,MAX
     1TIME,MLDIFPT,NIC,ERO
C
C        THIS SUBROUTINE SOLVES THE   VISCID X MOMENTUM EQUATION
C
      DIMENSION AM(101), BM(101), CM(101), DM(101), A(101), BETA(101), T
     1UMP(101), TIMP(101), AVM(101), BVM(101), CVM(101), DVM(101)
      DIMENSION DEL(101)
      TEMP=1./(2.*DELTA*DETA)
      NL1=N-1
      DELT=1./(2.*REY*DELTA*DETA*DETA)
      A(1)=0.
      NTIMES=0
      NXIT=10
      EPSX=.01
1     CONTINUE
      IF (NTIMES.LT.NXIT) GO TO 2
      WRITE (6,8)
      STOP
2     CONTINUE
      NTIMES=NTIMES+1.
      AM(1)=0.
      AM(N)=0.
      BM(1)=1.
      BM(N)=1.
      CM(1)=0.
      CM(N)=0.
      DM(1)=0.
      DM(N)=0.
      AVM(1)=0.
      AVM(N)=0.
      BVM(1)=1.
      BVM(N)=1.
      CVM(1)=0.
      CVM(N)=0.
      DVM(1)=0.
      DVM(N)=0.
      TUMP(1)=0.
      TIMP(1)=0.0
      DO 3 I=2,NL1
      CM(I)=-ROV(I)*TEMP*SCI(I)*RO(I)
      AM(I)=-CM(I)
3     CONTINUE
      DO 4 I=1,N
4     TUMP(I)=1./RO(I)
      CALL TINT (N,DETA,TUMP,TIMP)
      DO 5 I=1,N
      ROMU(I)=RO(I)*EMU(I)
5     DEL(I)=DELT*SCI(I)*RO(I)
      DO 6 I=2,NL1
      BM(I)=2.*RO(I)*A(I)-ROV(I)*Q
      DM(I)=BETA(I)-RO(I)*A(I)**2+ROV(I)*(A(I+1)-A(I-1))*TEMP*SCI(I)*RO(
     1I)+ROV(I)*A(I)*Q
      AVM(I)=-(ROMU(I-1)+ROMU(I))*DEL(I)/DELTA
      BVM(I)=(ROMU(I-1)+2.*ROMU(I)+ROMU(I+1))*DEL(I)/DELTA
      CVM(I)=-(ROMU(I)+ROMU(I+1))*DEL(I)/DELTA
      DVM(I)=-AVM(I)*A(I-1)-BVM(I)*A(I)-CVM(I)*A(I+1)
      AM(I)=AM(I)+AVM(I)
      BM(I)=BM(I)+BVM(I)
      CM(I)=CM(I)+CVM(I)
      DM(I)=DM(I)+DVM(I)
6     CONTINUE
```

```
      CALL TRIDIAG (N,AM,BM,CM,DM)
      IGOR=0
      DO 7 I=1,N
      A(I)=A(I)+DM(I)
      IF (ABS(DM(I)).LT.EPSX) GO TO 7
      IGOR=IGOR+1
7     CONTINUE
      IF (IGOR.NE.0) GO TO 1
      RETURN
C
C
8     FORMAT (10X,28HX MOMENTUM FAILS TO CONVERGE)
      END
```

```
      SUBROUTINE ENERGY
C
C     WINDWARD DIFFERENCE FORM OF ENERGY EQUATION
C
C     RO(I)IS UPDATED IN THIS SUBROUTINE BEFORE H IS UPDATED
      COMMON /CA/ A(101),B(101)
      COMMON /CY/ Y(101),YOYS(101)
      COMMON /CRO/ RO(101)
      COMMON /CETA/ ETA(101)
      COMMON /CV/ V(101)
      COMMON /CP/ P(101)
      COMMON /CT/ T(101)
      COMMON /CROV/ ROV(101)
      COMMON /CDV/ DV(101)
      COMMON /CSCH/ SCH(101),SCI(101)
      COMMON /CTEMP/ TEMP(101),TIMP(101)
      COMMON /CEMU/ EMU(101)
      COMMON /CROMU/ ROMU(101)
      COMMON /CPRAN/ PRAN(101)
      COMMON /CCAPH/ CAPH(101)
      COMMON /CH/ H(101)
      COMMON /CQR/ QR(101)
      COMMON /CDQR/ DQR(101)
      COMMON /CXH/ XNH(101)
      COMMON /RADCOMP/ DELBA,IRFLUX
      COMMON /FRSTRM/ HIN,PIN,RHOINP,UINP,RE
      COMMON /MISCL/ N,IDIM,CURV,DETA,DELTA,PSHOCK,REY,HW,EP,EROV,EA,MAX
     1TIME,MLDIFPT,NIC,ERO
      COMMON /DAMP/ IRODAMP,RODAMP,HDAMP
      DIMENSION AM(101), BM(101), CM(101), DM(101)
      DIMENSION VIS(101), CONV(101), VG(101), SCHS(101)
      DIMENSION DVM(101), DQRM(101)
      CALL RADARAY
      DO 1 I=1,N
      ROMU(I)=RO(I)*EMU(I)
1     CONTINUE
      DO 2 I=1,N
      SCHS(I)=SCH(I)*SCI(I)
      VIS(I)=SCHS(I)*ROMU(I)/(PRAN(I)*DETA)
      CONV(I)=DELTA*REY*SCHS(I)*ROV(I)
      VG(I)=VIS(I)*ROV(I)/RO(I)
2     CONTINUE
      NLI=N-1
      DO 3 I=2,NLI
      SIGN=0.
      IF (ROV(I).GT.0.) SIGN=1.0
      AM(I)=VIS(I-1)+VIS(I)-2.*CONV(I)*(1.-SIGN)
      BM(I)=-(VIS(I-1)+2.*VIS(I)+VIS(I+1))
      BM(I)=BM(I)+2.*CONV(I)*(1.-SIGN)-2.*CONV(I)*SIGN
      CM(I)=VIS(I)+VIS(I+1)+2.*CONV(I)*SIGN
      DVM(I)=(VG(I-1)+VG(I))*V(I-1)-(VG(I-1)+2.*VG(I)+VG(I+1))*V(I)+(VG(
     1I)+VG(I+1))*V(I+1)
      DQRM(I)=DELTA*REY*(-SCHS(I)*QR(I-1)+(-SCHS(I-1)+SCHS(I+1))*QR(I)+S
     1CHS(I)*QR(I+1))
      DM(I)=DVM(I)+DQRM(I)
3     CONTINUE
      AM(1)=0.
      AM(N)=0.
      BM(1)=1.
      BM(N)=1.
      CM(1)=0.
      CM(N)=0.
      DM(1)=HW+V(1)*V(1)/2.
      DM(N)=HIN
      CALL TRIDIAG (N,AM,BM,CM,DM)
      DO 4 I=1,N
      CAPH(I)=DM(I)
      XNH(I)=CAPH(I)-V(I)*V(I)/2.
      H(I)=HDAMP*H(I)+(1.-HDAMP)*XNH(I)
4     CONTINUE
```

```
      DO 6 I=1,N
      IF (H(I).LT.H(1)) 5,6
5     CONTINUE
      WRITE (6,7)
      WRITE (6,8) ETA(I),H(I)
      H(I)=H(1)
6     CONTINUE
      RETURN
C
C
7     FORMAT (1H0,5X,19HH(I) LESS THAN H(1))
8     FORMAT (2E14.6)
      END
```

```
      SUBROUTINE RADARAY
C
C     RADARAY GIVES THE RADIATION HEAT FLUX AT NIC POINTS
C     AND LINEARLY INTERPOLATES FOR THE REMAINING N POINTS
C
C
C     N MUST BE LESS THAN 100
C
      COMMON /CRO/ RO(101)
      COMMON /CETA/ ETA(101)
      COMMON /CY/ Y(101),YOYS(101)
      COMMON /CP/ P(101)
      COMMON /CH/ H(101)
      COMMON /CQR/ QR(101)
      COMMON /CXRO/ XNRO(101)
      COMMON /CT/ T(101)
      COMMON /CAMFI/ AMFC(100),AMFN(100),AMFO(100),AMFH(100)
      COMMON /RADCOMP/ DELBA,IRFLUX
      COMMON /MISCL/ N,IDIM,CURV,DETA,DELTA,PSHOCK,REY,HW,EP,EROV,EA,MAX
     1TIME,MLDIFPT,NIC,ERO
      COMMON /FRSTRM/ HIN,PIN,RHOINP,UINP,RB
      COMMON /DAMP/ IRODAMP,RODAMP,HDAMP
      DIMENSION PH(100), HH(100), ETAH(100), PHP(100), HHP(100), QRHP(10
     10), RHOHP(100), NICN(100)
      DIMENSION ETAN(100), QRP(100)
      NN=N
      DO 1 I=1,NN
      ETAH(I)=Y(I)/Y(N)
      PH(I)=P(I)
      HH(I)=H(I)
      CALL DIMPROP (PH(I),HH(I),PHP(I),HHP(I))
1     HHP(I)=HHP(I)*(.2389E-07)
      NRAD=(NN-1)/(NIC-1)
      DO 2 I=1,NIC
      J=NRAD*(I-1)+1
      ETAN(I)=ETAH(J)
2     CONTINUE
      DO 3 I=1,NIC
3     NICN(I)=NRAD*(I-1)+1
      IF (IRFLUX.LE.0) GO TO 6
      CALL RATRAP (DELBA,NIC,NICN,NN,4000.0,ETAH,PHP,AMFC,AMFN,AMFO,HHP,
     1QRHP,1,2,RHOHP,1,T)
      DO 4 I=1,NN
      CALL FTLUP (ETAH(I),QRP(I),1,NIC,ETAN,QRHP)
4     CONTINUE
      DO 5 I=1,N
5     QR(I)=QRP(I)/(RHOINP*UINP*UINP*UINP)*1.E+07
      GO TO 8
6     CONTINUE
      CALL RATRAP (DELBA,NIC,NICN,NN,4000.0,ETAH,PHP,AMFC,AMFN,AMFC,HHP,
     1QRHP,0,2,RHOHP,1,T)
      DO 7 I=1,N
7     QR(I)=0.
8     CONTINUE
      IF (IRODAMP.GT.0) GO TO 10
      DO 9 I=1,N
9     RO(I)=RHOHP(I)/RHOINP
      GO TO 13
10    CONTINUE
      DO 11 I=1,N
11    XNRO(I)=RHOHP(I)/RHOINP
      DO 12 I=1,N
12    RO(I)=RODAMP*RO(I)+(1.-RODAMP)*XNRO(I)
13    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE PRINTIT
C
C
C     N MUST BE LESS THAN 100
C
      COMMON /CRO/ RO(101)
      COMMON /CETA/ ETA(101)
      COMMON /CY/ Y(101),YOYS(101)
      COMMON /CP/ P(101)
      COMMON /CH/ H(101)
      COMMON /CQR/ QR(101)
      COMMON /CXRO/ XNRO(101)
      COMMON /CT/ T(101)
      COMMON /CAMFI/ AMFC(100),AMFN(100),AMFO(100),AMFH(100)
      COMMON /RADCOMP/ DELBA,IRFLUX
      COMMON /MISCL/ N,IDIM,CURV,DETA,DELTA,PSHOCK,REY,HW,EP,EROV,EA,MAX
     1TIME,MLDIFPT,NIC,ERO
      COMMON /FRSTRM/ HIN,PIN,RHOINP,UINP,RB
      COMMON /DAMP/ IRODAMP,RODAMP,HDAMP
      DIMENSION PH(100), HH(100), ETAH(100), PHP(100), HHP(100), QRHP(10
     10), RHOHP(100), NICN(100)
      DIMENSION ETAN(100), QRP(100)
      NN=N
      DO 1 I=1,NN
      ETAH(I)=Y(I)/Y(N)
      PH(I)=P(I)
      HH(I)=H(I)
      CALL DIMPROP (PH(I),HH(I),PHP(I),HHP(I))
1     HHP(I)=HHP(I)*(.2389E-07)
      NRAD=(NN-1)/(NIC-1)
      DO 2 I=1,NIC
      J=NRAD*(I-1)+1
      ETAN(I)=ETAH(J)
2     CONTINUE
      DO 3 I=1,NIC
3     NICN(I)=NRAD*(I-1)+1
      IF (IRFLUX.LE.0) GO TO 6
      CALL RATRAP (DELBA,NIC,NICN,NN,4000.0,ETAH,PHP,AMFC,AMFN,AMFO,HHP,
     1QRHP,1,2,RHOHP,0,T)
      WRITE (6,9)
      DO 4 I=1,NN
      CALL FTLUP (ETAH(I),QRP(I),2,NIC,ETAN,QRHP)
      WRITE (6,10) ETA(I),Y(I),YOYS(I),QRP(I)
4     CONTINUE
      DO 5 I=1,N
5     QR(I)=QRP(I)/(RHOINP*UINP*UINP*UINP)*1.E+07
      GO TO 8
6     CONTINUE
      CALL RATRAP (DELBA,NIC,NICN,NN,4000.0,ETAH,PHP,AMFC,AMFN,AMFO,HHP,
     1QRHP,0,2,RHOHP,0,T)
      DO 7 I=1,N
7     QR(I)=0.
8     CONTINUE
      RETURN
C
C
9     FORMAT (4X,6HETA(I),9X,4HY(I),8X,7HYOYS(I),3X,15HQRAD(I),W/SQ CM)
10    FORMAT (4E14.6)
      END
```

```
      FUNCTION D2DP2 (X)
      COMMON /CD2DP2/ D2DP2C
      D2DP2=D2DP2C
      RETURN
      END
```

```
      SUBROUTINE DIFT (Y,X,DY,N)
      DIMENSION Y(101), X(101), DY(101)
      DY(1)=(Y(2)-Y(1))/(X(2)-X(1))
      DY(N)=(Y(N)-Y(N-1))/(X(N)-X(N-1))
      N1=N-1
      DO 1 I=2,N1
      X1=X(I-1)-X(I)
      X3=X(I+1)-X(I)
      DY(I)=((Y(I+1)-Y(I))*X1**2+(Y(I)-Y(I-1))*X3**2)/((X1-X3)*X1*X3)
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE TRIDIAG (N,A,B,C,D)
      DIMENSION A(201), B(201), C(201), D(201)
C
C      THIS ROUTINE SOLVES AX=D FOR A TRIDIAGONAL
C
      N1=N-1
      C(1)=C(1)/B(1)
      D(1)=D(1)/B(1)
      DO 1 M=2,N1
      D(M)=(D(M)-A(M)*D(M-1))/(B(M)-A(M)*C(M-1))
    1 C(M)=C(M)/(B(M)-A(M)*C(M-1))
      D(N)=(D(N)-A(N)*D(N-1))/(B(N)-A(N)*C(N-1))
      DO 2 J=2,N
      M=N+1-J
    2 D(M)=D(M)-C(M)*D(M+1)
      RETURN
      END
```

```
      SUBROUTINE SET (N,A,B)
      DIMENSION A(101), B(101)
      DO 1 I=1,N
    1 B(I)=A(I)
      RETURN
      END
```

```
      FUNCTION ITEST (N,A,B,E)
      DIMENSION A(101), B(101)
      ITEST=0
      DO 1 J=1,N
    1 IF (ABS(A(J)-B(J)).GT.E) ITEST=ITEST+1
      RETURN
      END
```

```
      SUBROUTINE DIFCOEF
      COMMON /CETA/ ETA(101)
      COMMON /CP/ P(101)
      COMMON /CH/ H(101)
      COMMON /CT/ T(101)
      COMMON /CD12/ D12(101)
      COMMON /MISCL/ N,IDIM,CURV,DETA,DELTA,PSHOCK,REY,HW,EP,EROV,EA,MAX
     1TIME,MLDIFPT,NIC,ERO
      COMMON /FRSTRM/ HIN,PIN,RHOINP,UINP,RB
      COMMON /COING11/ TSTR(18),OMEGA11(18)
      DIMENSION TST(101), OMEG(101)
      DIMENSION PHP(101), HHP(101)
      REAL M1,M2,M12
C     BINARY DIFFUSION COEFFICIENT
C     1-ATOMIC NITROGEN
C     2-ATOMIC HYDROGEN
      M1=14.01
      M2=1.008
      SIGMA1=3.298
      SIGMA2=2.708
      EPSOK1=71.4
      EPSOK2=37.0
      SIGMA12=.5*(SIGMA1+SIGMA2)
      EPSOK12=SQRT(EPSOK1*EPSOK2)
      DO 1 I=1,N
      TST(I)=T(I)/EPSOK12
      CALL FTLUP (TST(I),OMEG(I),-1,18,TSTR,OMEGA11)
    1 CONTINUE
      DO 2 I=1,N
      CALL DIMPROP (P(I),H(I),PHP(I),HHP(I))
    2 CONTINUE
      M12=(M1+M2)/(2.*M1*M2)
      M12=SQRT(M12)
      DC=0.002628*M12/(SIGMA12*SIGMA12)
      DO 3 I=1,N
      D12(I)=DC/(PHP(I)*OMEG(I))*(T(I))**1.5/(RB*UINP)
    3 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE VISPRAN
      COMMON /CP/ P(101)
      COMMON /CH/ H(101)
      COMMON /CT/ T(101)
      COMMON /CEMU/ EMU(101)
      COMMON /CPRAN/ PRAN(101)
      COMMON /CETA/ ETA(101)
      COMMON /FRSTRM/ HIN,PIN,RHOINP,UINP,RB
      COMMON /MISCL/ N,IDIM,CURV,DETA,DELTA,PSHOCK,REY,HW,EP,EROV,EA,MAX
     1TIME,MLDIFPT,NIC,ERO
      COMMON /REFVALU/ EMUSHOK
      DIMENSION PP(101), HP(101), TP(101)
      EXTERNAL VISCOS
      EXTERNAL PRANDTL
      DO 1 I=1,N
      TP(I)=T(I)
      CALL DIMPROP (P(I),H(I),PP(I),HP(I))
      EMU(I)=VISCOS(TP(I),PP(I))
      EMU(I)=EMU(I)/EMUSHOK
      PRAN(I)=PRANDTL(TP(I),PP(I))
    1 CONTINUE
      RETURN
      END
```

```
      FUNCTION RHOSY (P,H,AFC,AFN,AFO,TGUES)
C
C         P ATMOSPHERES,H DEGREES K, RHOSY AMAGATS
C
      COMMON /FRSTRM/ HIN,PIN,RHOINP,UINP,RB
C
C     INPUTS TO RHOSY P ATMOSPHERES,H DEGREES K
C     INPUTS TO RATRAP P ATMOSPHERES,H CAL/GM
C     OUTPUT FROM RATRAP RHOSY GM/CC
C
      CALL RATRAP (1.,1,1,1,TGUES,0.,P,AFC,AFN,AFO,H/(3.483*4.19),QR,0,2
     1,RHOSY,1,T)
      TGUES=T
      RHCSY=RHOSY/RHOINP
      RETURN
      END
```

```
      FUNCTION VISCOS (AT,AP)
      COMMON /HANSEN/ TABT(31),TABPLOG(6),TABZH(186),TABZ(186),TABPR(186
     1),TEMURAT(186)
      EXTERNAL RHOSY
C     INPUT T TEMPERATURE,DEG.K
C           P PRESSURE,ATM
C     OUTPUT VISCOSITY,GM/CM/SEC
      T=AT
      P=AP
      PLOG=ALOG10(P)
      CALL DISCOT (T,PLOG,TABT,TEMURAT,TABPLOG,21,186,6,VISCOS)
      IF (T.LE.15000.) GO TO 1
      WRITE (6,2) T
1     VISCOS=VISCOS*(1.462E-5)*SQRT(T)/(1.+112./T)
      RETURN
C
C
2     FORMAT (6X,25HTEMPERATURE TOO LARGE. T=E16.6,5HDEG.K)
      END
```

```
      FUNCTION PRANCTL (AT,AP)
      COMMON /HANSEN/ TABT(31),TABPLOG(6),TABZH(186),TABZ(186),TABPR(186
     1),TEMURAT(186)
C     INPUT T TEMPERATURE,DEG.K
C           P PRESSURE ATMOS
      T=AT
      P=AP
      PLOG=ALOG10(P)
      CALL DISCOT (T,PLOG,TABT,TABPR,TABPLOG,21,186,6,PRAN)
      PRANDTL=PRAN
      IF (T.LE.15000.) GO TO 1
      WRITE (6,2) T
1     RETURN
C
C
2     FORMAT (6X,25HTEMPERATURE TOO LARGE. T=E16.6,5HDEG.K)
      END
```

```
      BLOCK DATA
      COMMON /HANSEN/ TABT(31),TABPLOG(6),TABZH(186),TABZ(186),TABPR(186
     1),TEMURAT(186)
      COMMON /COING11/ TSTR(18),OMEGA11(18)
C     T TEMPERATURE  CEGREES K
C               P PRESSURE, ATM
      DATA TABT/100.,500.,1000.,1500.,2000.,2500.,3000.,3500.,4000.,4500
     1.,5000.,5500.,6000.,6500.,7000.,7500.,8000.,8500.,9000.,9500.,1000
     20.,10500.,11000.,11500.,12000.,12500.,13000.,13500.,14000.,14500.,
     315000./
      DATA TABPLOG/-4.,-3.,-2.,-1.,0.,1./
      DATA TABZH/3.52,3.52,3.65,3.80,4.41,8.02,8.19,8.03,9.82,16.80,23.4
     16,23.58,22.54,21.93,22.29,24.26,28.65,35.75,43.74,49.15,50.96,50.6
     24,49.48,48.07,46.64,45.26,43.97,42.76,41.64,40.58,39.60,3.52,3.52,
     33.65,3.80,4.07,6.16,8.02,7.77,8.09,10.55,16.68,21.58,21.97,21.24,2
     40.69,20.72,21.65,23.85,27.66,33.00,38.79,43.28,45.57,46.09,45.64,4
     54.74,43.69,42.61,41.55,40.53,39.57,2*3.52,3.65,3.80,3.97,4.81,7.13
     6,7.62,7.53,8.14,10.48,15.14,19.30,20.35,20.01,19.54,19.34,19.60,20
     7.49,22.17,24.78,28.28,32.31,36.13,39.01,40.66,41.26,41.17,40.69,40
     8.01,39.24,2*3.52,3.65,3.80,3.93,4.27,5.55,7.08,7.28,7.33,7.96,9.73
     9,12.93,16.46,18.34,18.66,18.43,18.17,18.09,18.29,18.85,19.84,21.31
     $,23.28,25.69,28.36,30.99,33.27,34.97,36.02,36.53,2*3.52,3.65,3.80,
     $3.92,4.09,4.61,5.75,6.74,6.98,7.10,7.58,8.70,10.64,13.20,15.48,16.
     $73,17.09,17.04,16.91,16.84,16.90,17.13,17.57,18.24,19.16,20.32,21.
     $72,23.29,24.98,26.66,2*3.52,3.65,3.80,3.92,4.03,4.25,4.75,5.56,6.2
     $9,6.62,6.80,7.11,7.72,8.76,10.24,11.99,13.63,14.79,15.40,15.61,15.
     $64,15.60,15.58,15.62,15.74,15.96,16.28,16.71,17.26,17.92/
      DATA TABZ/4*1.,1.016,1.163,1.2,1.211,1.287,1.577,1.91,1.99,2.008,2
     1.032,2.088,2.21,2.446,2.826,3.282,3.645,3.843,3.932,3.969,3.985,3.
     2993,3.996,3.998,2*3.999,2*4.,4*1.,1.005,1.088,1.192,1.203,1.228,1.
     3337,1.622,1.898,1.983,2.006,2.027,2.067,2.144,2.284,2.51,2.832,3.2
     402,3.526,3.745,3.867,3.931,3.963,3.979,3.988,3.993,3.996,3.997,4*1
     5.,1.002,1.033,1.149,1.197,1.208,1.245,1.359,1.599,1.849,1.961,1.99
     67,2.017,2.044,2.09,2.166,2.286,2.462,2.700,2.983,3.272,3.52,3.7,3.
     7818,3.889,3.932,3.957,3.973,4*1.,1.001,1.011,1.072,1.167,1.198,1.2
     813,1.252,1.348,1.529,1.752,1.904,1.971,2.001,2.023,2.05,2.09,2.149
     9,2.234,2.351,2.505,2.694,2.91,3.135,3.347,3.527,3.667,3.769,5*1.,1
     $.004,1.026,1.092,1.165,1.196,1.214,1.248,1.316,1.437,1.607,1.778,1
     $.896,1.959,1.993,2.018,2.042,2.071,2.111,2.163,2.232,2.318,2.426,2
     $.553,2.7,2.861,3.028,5*1.,1.001,1.009,1.035,1.089,1.149,1.186,1.20
     $8,1.235,1.279,1.351,1.457,1.59,1.727,1.838,1.914,1.962,1.993,2.018
     $,2.042,2.067,2.098,2.135,2.18,2.233,2.297,2.372/
      DATA TABPR/2*.738,.756,.767,.614,.771,.714,.606,.587,.764,.993,.87
     11,.455,.392,.361,.342,.322,.279,.2,.114,.0576,.0314,.0213,.0167,.0
     2143,.0129,.0121,.011,.0108,.0109,.011,2*.738,.756,.767,.668,.654,.
     3745,.658,.58,.611,.799,.989,.891,.464,.404,.371,.351,.335,.316,.27
     49,.216,.145,.0877,.0524,.0346,.0238,.019,.0162,.0149,.013,.012,2*.
     5738,.756,.767,.724,.611,.74,.737,.619,.578,.624,.785,.969,.955,.83
     6,.424,.387,.363,.348,.336,.319,.295,.254,.201,.146,.101,.0688,.047
     7,.0345,.0245,.0129,2*.738,.756,.767,.766,.645,.636,.744,.759,.61,.
     8581,.617,.736,.906,.986,.969,.648,.411,.382,.364,.348,.339,.327,.3
     912,.292,.263,.227,.185,.144,.0986,.0819,2*.738,.756,.767,.773,.696
     $,.627,.660,.762,.752,.611,.583,.602,.673,.796,.927,.983,.943,.807,
     $.497,.429,.404,.382,.369,.355,.343,.333,.319,.302,.277,.253,2*.738
     $,.756,.767,.773,.751,.680,.631,.662,.743,.767,.62,2*.592,.62,.688,
     $.788,.891,.961,.966,.872,.532,.463,.434,.412,.396,.383,.369,.36,.3
     $49,.341/
      DATA TEMURAT/7*1.,1.011,1.032,1.096,1.181,1.227,1.256,1.271,1.264,
     11.21,1.072,.826,.517,.261,.118,.055,.029,.018,.012,.009,.008,2*.00
     27,2*.008,7*1.,1.01,1.024,1.055,1.128,1.209,1.257,1.286,1.303,1.307
     3,1.28,1.207,1.068,.853,.595,.361,.2,.108,.063,.036,.024,.018,.015,
     4.013,.012,7*1.,1.01,1.022,1.038,1.074,1.146,1.228,1.276,1.317,1.33
     57,1.347,1.343,1.314,1.251,1.143,.983,.782,.571,.387,.249,.158,.1,.
     6067,.042,.016,7*1.,1.006,1.02,1.033,1.051,1.086,1.148,1.229,1.294,
     71.332,1.371,1.386,1.396,1.393,1.375,1.335,1.267,1.168,1.04,.881,.7
     811,.547,.408,.262,.212,7*1.,1.003,1.016,1.029,1.043,1.06,1.09,1.13
     99,1.208,1.283,1.342,1.386,1.425,1.438,1.445,1.448,1.442,1.424,1.39
     $4,1.342,1.274,1.187,1.082,.94,.828,7*1.,1.001,1.008,1.022,1.036,1.
     $052,1.067,1.09,1.124,1.175,1.238,1.307,1.368,1.418,1.468,1.496,1.5
     $01,1.511,1.52,1.516,1.508,1.492,1.468,1.415,1.387/
```

```
      DATA TSTR/5.,6.,7.,8.,9.,10.,20.,30.,40.,50.,60.,70.,80.,90.,100.,
     1200.,300.,400./
      DATA OMEGA11/.8422,.8124,.7896,.7712,.7556,.7424,.6640,.6232,0.596
     10,0.5756,0.5596,0.5464,0.5352,0.5256,0.5170,0.4644,0.4360,0.4170/
      END
```

```
      SUBROUTINE DIMPROP (AP,AH,PPRIME,HPRIME)
      COMMON /FRSTRM/ HIN,PIN,RHOINP,UINP,RB
C
C     INPUTS TO DIMPROP ARE DIMENSIONLESS
C     OUTPUTS P ATOMOSPHERES ,H ERGS/GM
C
      PPRIME=AP*RHOINP*UINP*UINP/(1.01325E+06)
      HPRIME=AH*UINP*UINP
      RETURN
      END
```

```
      SUBROUTINE TINT (N,DX,A,B)
C         THIS SUBROUTINE INTEGRATES A TABLE TO GIVE A TABLE
      DIMENSION A(102), B(102)
      NHAF=N/2
      IEVEN=0
      IF (2*NHAF.EQ.N) IEVEN=1
      IF (IEVEN.EQ.1) NHAF=NHAF-1
      B(1)=0.
      DO 1 I=1,NHAF
      J=2*I
      B(J)=B(J-1)+.08333333*DX*(5.*A(J-1)+8.*A(J)-A(J+1))
      J=J+1
      B(J)=B(J-2)+.33333333*DX*(A(J-2)+4.*A(J-1)+A(J))
1     CONTINUE
      IF (IEVEN.EQ.0) GO TO 2
      J=J+1
      B(J)=B(J-1)+.08333333*DX*(-A(J-2)+8.*A(J-1)+5.*A(J))
2     RETURN
      END
```

```
      SUBROUTINE GET REDY
      DIMENSION PALH(100),PALC(100),PALN(100),PALO(100)
      DIMENSION  A(6,6)     ,A12(16,7) ,A22(7,7)  ,ALP(6)     ,ALPT(16,5),FEMP0040
     1BMT(16)    ,C(6,16)   ,CH(16,2)  ,CP(16)     ,DEGI(6)    ,H(16)     ,FEMP0050
     2JAT(16,5) ,JPH(16)    ,KAT(6)    ,KODE(16)   ,RA(16,2)   ,RB(16,2)  ,FEMP0060
     3RC(16,2)  ,RD(16,2)   ,RD1(16,2) ,RE(16,2)   ,RE1(16,2) ,SD(16)     ,FEMP0070
     4TB(3)     ,TU(16,2)   ,TU2(16,2) ,VN(17)     ,VNE(16)    ,VNT(17)   ,FEMP0080
     5VNU(16,6) ,W26(6)     ,W3(16)    ,Y(16)      ,RF(16,2)   ,RC1(16,2)  FEMP0090
      DIMENSION SAC(200), SAH(200)
      COMMON /LOKHEED/
     A          A      ,A12    ,A22    ,ALP    ,ALPT   ,BMT    ,C      ,CH     ,  FEMP0100
     1CP     ,DEGI   ,H      ,HS     ,IG     ,IGMS   ,IGMSP  ,IGP    ,ION    ,  FEMP0110
     2IS     ,ISP    ,ISPNG  ,ISPNGP,ISPNG2,JAT    ,JPH    ,KAT    ,KODE   ,  FEMP0120
     3N      ,NG     ,NP     ,PRESS  ,RA     ,RB     ,RC     ,RD     ,RD1    ,  FEMP0130
     4RE     ,RE1    ,SD     ,TU     ,TU2    ,VN     ,VNE    ,VNT    ,VNU    ,  FEMP0140
     5W26    ,W27    ,W3     ,Y      ,RHO    ,WM     ,SYU    ,RF     ,RC1,TB     FEMP0150
      COMMON/RAD/YY(100),TEE(100),FHV(20),NHV,NY,C2,IY
      COMMON/RAD/QRYPC(100),QRYPL(100)
      COMMON/RAD/XNN(14,100),XMOL
      COMMON/RAD/ NIHVC,FHVC(50),AHV(50),AHVL(20)
      COMMON /RAD/C1,C3,C4,FLG,C5,FLG1
      COMMON /RAD/ YDELT,DELTA,FL1,FL2
      COMMON/RAD/GEE(8),EPS(8),NU(20),ND(70),FF(70),GAMP(70),
     1WOL(20),FHVM(20),FHVP(20)
      COMMON/CIONCL/F(100,10),F2(100,10),HVL(70),EP,K2,K1,IFL,IYCON,
     1IQ1,WMI,BIJ(100,10),GMIN(100,10),GPLU(100,10) ,IAED
      DIMENSION  NICN(100)
      DIMENSION PRES(100)
      DIMENSION HH(100)
      DIMENSION ZBLK(1157)
      DIMENSION QRYP(100)
      EQUIVALENCE (ZBLK,A)
      XMOL=1.
      DO 1157 I=1,1157
 1157 ZBLK(I)=0.
      NHV=18
      DO 998 I=1,17
  998 VNT(I)=0.
      C4=1.273
      G=1.
  100 FORMAT(6E12.1)
C
C     INPUT TABLES I,II
C
      READ(5,100) (GEE(I),I=1,8)
      READ(5,100) (EPS(I),I=1,8)
      READ(5,100)  (FHVM(I),I=1,NHV)
      READ(5,100)  (FHVP(I),I=1,NHV)
      READ(5,100)  ( FHV(I),I=1,NHV)
      READ(5,100) (WOL(I),I=1,NHV)
      READ(5,101)  (   NU(I),I=1,NHV)
  101 FORMAT(18I2)
      IS=0
      DO 1 I=1,NHV
    1 IS=IS+NU(I)
      READ(5,102) (ND(I),HVL(I),FF(I),GAMP(I),I=1,IS)
  102 FORMAT(I1,11X,3E12.1)
C
C     INPUT HV VALUES AND AK FACTORS FOR CONTM CALCULATION
C
      READ(5,115) NIHVC
  115 FORMAT(24I3)
      READ(5,100) (FHVC(I),I=1,NIHVC)
      READ(5,100) (AHV(I),I=1,NIHVC)
      READ (5,100) (AHVL(I),I=1,NHV)
      RETURN
      END
```

```
      SUBROUTINE RATRAP ( DALTA, NIC, NICN, NX, TFX, YX, PRES, PALC,
     1PALN,PALO,HH,QRYP,TRAD,IOPT,RO,KUT,TEEE)
      DIMENSION PALH(100),PALC(100),PALN(100),PALO(100)
      DIMENSION  A(6,6)      ,A12(16,7) ,A22(7,7)   ,ALP(6)      ,ALPT(16,5),FEMP0040
     1BMT(16)    ,C(6,16)    ,CH(16,2)  ,CP(16)     ,DEGI(6)     ,H(16)     ,FEMP0050
     2JAT(16,5)  ,JPH(16)    ,KAT(6)    ,KODE(16)   ,RA(16,2)    ,RB(16,2)  ,FEMP0060
     3RC(16,2)   ,RD(16,2)   ,RD1(16,2) ,RE(16,2)   ,RE1(16,2)   ,SD(16)    ,FEMP0070
     4TB(3)      ,TU(16,2)   ,TU2(16,2) ,VN(17)     ,VNE(16)     ,VNT(17)   ,FEMP0080
     5VNU(16,6)  ,W26(6)     ,W3(16)    ,Y(16)      ,RF(16,2)    ,RC1(16,2)  FEMP0090
      DIMENSION SAC(200), SAH(200)
      COMMON /LOKHEED/
     A      A      ,A12    ,A22    ,ALP    ,ALPT   ,BMT    ,C      ,CH     ,  FEMP0100
     1CP    ,DEGI  ,H      ,HS     ,IG     ,IGMS   ,IGMSP  ,IGP    ,ION    ,  FEMP0110
     2IS    ,ISP   ,ISPNG  ,ISPNGP,ISPNG2,JAT    ,JPH    ,KAT    ,KODE   ,  FEMP0120
     3N     ,NG    ,NP     ,PRESS  ,RA     ,RB     ,RC     ,RD     ,RD1    ,  FEMP0130
     4RE    ,REI   ,SD     ,TU     ,TU2    ,VN     ,VNE    ,VNT    ,VNU    ,  FEMP0140
     5W26   ,W27   ,W3     ,Y      ,RHO    ,WM     ,SYU    ,RF     ,RC1,TB    FEMP0150
      COMMON/RAD/YY(100),TEE(100),FHV(20),NHV,NY,C2,IY
      COMMON/RAD/QRYPC(100),QRYPL(100)
      COMMON/RAD/XNN(14,100),XMOL
      COMMON/RAD/ NIHVC,FHVC(50),AHV(50),AHVL(20)
      COMMON /RAD/C1,C3,C4,FLG,C5,FLG1
      COMMON /RAD/ YDELT,DELTA,FL1,FL2
      COMMON/RAD/GEF(8),EPS(8),NU(20),ND(70),FF(70),GAMP(70),
     1WOL(20),FHVM(20),FHVP(20)
      COMMON/CIONCL/F(100,10),F2(100,10),HVL(70),EP,K2,K1,IFL,IYCON,
     1IQ1,WMI,BIJ(100,10),GMIN(100,10),GPLU(100,10) ,IAED
      DIMENSION  NICN(100)
      DIMENSION PRES(100)
      DIMENSION HH(100)
      DIMENSION ZBLK(1157)
      DIMENSION QRYP(100)
      DIMENSION YX(100)
      DIMENSION RO(100)
      DIMENSION X(25)
      DIMENSION  XNC2H(40),XNC3H(40),XNC4H(40),XNHCN(40),XNC2H2(40),
     1XNHP(40)
      DIMENSION TEEE(100)
      EQUIVALENCE (ZBLK,A)
      DELTA=DALTA
      NY=NX
      DO 200 I=1,NY
  200 YY(I)=YX(I)
      FL1= 2.
      FL2= 2.
      FLG= 0.
      FLG1=1.
      TB(1)=TFX
      LONGRIT=0
      PUT12=100./12.
      PUT14=100./14.
      PUT16=100./16.
      DO 225 I=1,NY
      SAC(I)= PALC(I)
      SAH(I)=1.0-(PALC(I)+PALN(I)+PALO(I))
      PALC(I)=PUT12*PALC(I)
      PALN(I)=PUT14*PALN(I)
      PALO(I)=PUT16*PALO(I)
      PALH(I)=100.-(12.*PALC(I)+14.*PALN(I)+16.*PALO(I))
      IF (PALC(I).LT.1.0E-03) PALH(I)=0.0
  225 CONTINUE
    4 DO 20 I=1,NY
      PRESS=PRES(I)
      IF(IOPT.EQ.2) GO TO 7
C
C     GIVEN T, GET H
C
      TB(2)=TB(1)
      GO TO 8
C
```

```
C     GIVEN H, GET T
    7 TB(2)=TB(1)*1.2
      HS=HH(I)
    3 I1=0
   12 CONTINUE
      IF (TB(2).LT.6000.) GO TO 9
C
C     HIGH TEMPERATURE CALCULATION
C
      ALP(2)=PALH(I)
      ALP(3)=PALC(I)
      ALP(4)=PALN(I)
      ALP(5)=PALO(I)
      ALP(1)=ALP(3) + ALP(4) + ALP(5) +ALP(2)
      IT=1
      ITXV=-1
      CALL HTEST (ITX,ITXV,1)
      ITXV=ITX
      CALL FEMP (IHELP,IOPT)
      CALL HTEST (ITX,ITXV,2)
      IZW=2
      GO TO 10
C
C     LOW TEMPERATURE
C
    9 ITSV=-1
      IT=1
      ALP(1)=PALH(I)
      ALP(2)=PALC(I)
      ALP(3)=PALN(I)
      ALP(4)=PALO(I)
      CALL LTEST (ITS,ITSV,1)
      ITSV=ITS
      CALL FEMP (IHELP,IOPT)
      CALL LTEST (ITS,ITSV,2)
      IZW=1
   10 IF (IOPT.EQ.2) GO TO 32
      TB(2)=TEE(I)
      TB(1)=TB(2)
      GO TO 33
   32 TEE(I)=TB(2)
      HS=HH(I)
      TB(1)=TB(2)/1.2
   33 IF(IHELP.EQ.0) GO TO 11
C
C     ERROR IN FEMP
C
      WRITE(6,110)I,PRESS,HS,TB(1),TB(2)
  110 FORMAT(10H1FEMP BLEW,I10,4E20.6)
      STOP
C
C     BACK TO (12) JUST ONE TIME...LOOP COULD OCCURE OTHERWISE
C
   11 IF(I1.NE.0) GO TO 13
      I1=1
      GO TO 12
C
C     CHECK FOR HIGH OR LOW TEMP CALCULATION
C     ...THEN COMPUTE NO. DENSITIES
C
   13 IF(IZW.EQ.2) GO TO 15
      XNHP(I)=0.0
      XNC2H(I)=FLUT(BMT(9),W3(9),RHO)
      XNC3H(I)=FLUT(BMT(10),W3(10),RHO)
      XNC4H(I)=FLUT(BMT(11),W3(11),RHO)
      XNHCN(I)=FLUT(BMT(12),W3(12),RHO)
      XNC2H2(I)=FLUT(BMT(13),W3(13),RHO)
      XNN(1,I)=0.0
      XNN(2,I)=  FLUT(BMT(6),W3(6),RHO)
```

```
      XNN(3,I)=0.0
      XNN(4,I)=  FLUT(BMT(8),W3(8),RHO)
      XNN(5,I)=0.0
      XNN(6,I)=  FLUT(BMT(3),W3(3),RHO)
      XNN(7,I)=  FLUT(BMT(4),W3(4),RHO)
      XNN(8,I)=  FLUT(BMT(15),W3(15),RHO)
      XNN(9,I)=  FLUT(BMT(14),W3(14),RHO)
      XNN(10,I)=  FLUT(BMT(2),W3(2),RHO)
      XNN(11,I)=  FLUT(BMT(1),W3(1),RHO)
      XNN(12,I)=  FLUT(BMT(7),W3(7),RHO)
      XNN(13,I)=0.0
      XNN(14,I)=FLUT(BMT(5),W3(5),RHO)
      GO TO 19
   15 XNN(1,I)=  FLUT(BMT(6),W3(6),RHO)
      XNHP(I)=FLUT(BMT(3),W3(3),RHO)
      XNC2H(I)=0.0
      XNC3H(I)=0.0
      XNC4H(I)=0.0
      XNHCN(I)=0.0
      XNC2H2(I)=0.0
      XNN(2,I)=  FLUT(BMT(8),W3(8),RHO)
      XNN(3,I)=   FLUT(BMT(7),W3(7),RHO)
      XNN(4,I)=  FLUT(BMT(5),W3(5),RHO)
      XNN(5,I)= FLUT(BMT(6),W3(6),RHO)+FLUT(BMT(7),W3(7),RHO)+
     1FLUT(BMT(11),W3(11),RHO)+XNHP(I)
      XNN(6,I)=0.0
      XNN(7,I)=  FLUT(BMT(9),W3(9),RHO)
      XNN(8,I)=  FLUT(BMT(10),W3(10),RHO)
      XNN(9,I)=0.0
      XNN(10,I)=0.0
      XNN(11,I)=  FLUT(BMT(4),W3(4),RHO)
      XNN(12,I)=  FLUT(BMT(2),W3(2),RHO)
      XNN(13,I)=  FLUT(BMT(11),W3(11),RHO)
      XNN(14,I)=FLUT(BMT(12),W3(12),RHO)
C
C     SET NEW GUESS FOR TEMP...APPLIES ONLY IF IOPT=2
C
   19 TB(2)=TEE(I)
C
      HH(I)=HS
      RO(I)=RHO
      IF ( KUT.EQ.1 ) GO TO 20
C     PRINT TEMP CALCULATIONS
C
      WRITE (6,109)
  109 FORMAT(14H0  PATH LENGTH,4X,
     18HPRESSURE,3X,11HTEMPERATURE,4X,8HENTHALPY,5X,7HDENSITY)
      WRITE(6,111) YY(I),PRES(I),TEE(I),HS,RHO
  111 FORMAT(10E13.5)
      XT=0.
      DO 2 J=1,14
    2 XT= XNN(J,I)+XT
      XT=XT+XNC2H(I)+XNC3H(I)+XNC4H(I)+XNHCN(I)+XNC2H2(I)+XNHP(I)
      DO 3 J=1,14
    3 X(J)= XNN(J,I)/XT
      X(15)=XNC2H(I)/XT
      X(16)=XNC3H(I)/XT
      X(17)=XNC4H(I)/XT
      X(18)=XNHCN(I)/XT
      X(19)=XNC2H2(I)/XT
      X(20)=XNHP(I)/XT
      WRITE (6,900) (X(J),J=1,20)
  900 FORMAT (5X14HMOLE FRACTIONS /6X,2HO+,10X,1HO,13X,2HN+,11X,1HN,11X,
     1 1HE,12X,2HO2,12X,2HN2 / 6X,2HCO,10X,2HH2,12X,2HC2,10X,2HCN,11X,
     2 1HC,12X,2HC+,12X,2H H/6X,3HC2H,9X,3HC3H,12X,3HC4H,10X,3HHCN,9X,
     3 4HC2H2,9X,2HH+/7E13.5/7E13.5/6E13.5)
   20 CONTINUE
      DO 1 I=1,NY
    1 TEEE(I)=TEE(I)
```

```
      IF (IRAD.EQ.0) GO TO 1938
C
C     INPUT AK FACTORS FOR LINE CALCULATIONS
C
      NW=MAX0(NHV,NIHVC)
  125 FORMAT(F12.3,F16.3)
  127 FORMAT(34X,F12.3,F16.3)
  128 FORMAT(2(F12.3,F16.3))
      IF(FLG1.EQ.0.) GO TO 60
C
C     MULT. FACTORS BY PI FOR FLUX CALCULATIONS
C
      DO 61 I=1,NHV
   61 AHVL(I)=AHVL(I)*3.1416
      DO 62 I=1,NIHVC
   62 AHV(I)=AHV(I)*3.1416
   60 C2=DELTA*FL1
      C1=DELTA*FL2
      C3=C1/3.1416
      C5=C1
      EP=.001
C
C     OUTPUT TABLE II
C
  120 FORMAT (1H1,60X,8HTABLE II)
  121 FORMAT (6H1GROUP,4X,2HHV,12X,3HHV+,11X,3HHV-,10X,1HN,9X,
     13HWOL,7X,4HK(I),4X,5HHV(I),10X,4HF(I),11X,6HGAM(I))
  122 FORMAT(I4,F12.3,2F14.3,I12,F12.3,I9,F14.3,1P2E16.2)
      IF (LONGRIT.EQ.0) GO TO 1066
      WRITE (6,121)
  124 FORMAT(68X,I9,F14.3,4X,1PE12.2,1PE16.2)
      IC1=0
      DO 30 I=1,NHV
      IC2 = IC1 + NU(I)
      IC1= IC1+1
      WRITE (6,122) I,FHV(I), FHVP(I), FHVM(I), NU(I), WOL(I),
     1ND(IC1),HVL(IC1),FF(IC1),GAMP(IC1)
      IF(IC1.EQ.IC2) GO TO 30
      IC3=IC1+1
      DO 25 J=IC3,IC2
   25 WRITE (6,124) ND (J),HVL(J), FF(J), GAMP(J)
   30 IC1=IC2
 1066 CONTINUE
      DO 300 L=1,NIC
      IY=NICN(L)
      IYCON=IY
      YDELT=YY(IY)
C
C     CONTINUM CALCULATION
       CALL CONTM
C
C     LINE CALCULATION
      CALL LINE
      ORYP(L)=ORYPC(IY)+ORYPL(IY)
  300 CONTINUE
      IF (FLG1.EQ.0.0) RETURN
      DO 434 I=1,NHV
  434 AHVL(I)=AHVL(I)/3.1416
      DO 435 I=1,NIHVC
  435 AHV(I)=AHV(I)/3.1416
 1938 CONTINUE
      DO 65 I=1,NY
      PALC(I)=.12*PALC(I)
      PALN(I)=.14*PALN(I)
      PALO(I)=.16*PALO(I)
   65 CONTINUE
      RETURN
      END
$SUB   1
```

APPENDIX B

SAMPLE INPUT

The input for a sample case is shown in this appendix as follows:

```
$NAM4
N         =  21,
IDIM      =  2,
IRODAMP   =  0,
HDAMP     =  0.7E+00,
RODAMP    =  0.7E+00,
EP        =  0.1E-01,
ERCV      =  0.1E-01,
EA        =  0.1E-01,
EPSX      =  0.1E-01,
ERO       =  0.2E-01,
MAXTIME   =  2,
NIC       =  11,
CURV      =  0.1E+01,
CURVZ     =  0.12E+01,
BS        =  0.1E+01,
D2PDZ     =  0.2E+01,
$END
```

```
$NAM5
IRFLUX    =  1,
IUPDATE   =  1,
ALPINJC   =  0.0,
ALPINJO   =  0.22E+00,
ALPINJH   =  0.0,
ALPINJN   =  0.78E+00,
D20P2C    =  0.25E+01,
AS        =  0.122E+01,
PINP      =  0.197E+03,
RHOINP    =  0.272E-06,
UINP      =  0.1525E+07,
RB        =  0.3048E+03,
ROVW      = -0.1E+00,
HW        =  0.28E-01,
ALPINFN   =  0.78E+00,
ALPINFO   =  0.22E+00,
ALPINFC   =  0.0,
ALPINFH   =  0.0,
$END
```

```
$NAM1
RO      =  0.195E+03,  0.182E+03,  0.17E+03,  0.157E+03,  0.143E+03,  0.13E+03,
           0.112E+03,  0.93E+02,  0.72E+02,  0.45E+02,  0.32E+02,  0.29E+02,
           0.27E+02,  0.271E+02,  0.257E+02,  0.245E+02,  0.234E+02,
           0.223E+02,  0.213E+02,  0.194E+02,  0.1704925E+02, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I,
ROV     = -0.1E+00, -0.99E-01, -0.976E-01, -0.95E-01, -0.909E-01, -0.852E-01,
          -0.777E-01, -0.68E-01, -0.548E-01, -0.391E-01, -0.149E-01,
           0.294E-01,  0.912E-01,  0.165E+00,  0.252E+00,  0.35E+00,  0.46E+00,
           0.58E+00,  0.71E+00,  0.851E+00,  0.1E+01, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I,
$END


$NAM2
A       =  0.0,  0.9E-02,  0.2E-01,  0.33E-01,  0.45E-01,  0.55E-01,  0.73E-01,
           0.9E-01,  0.115E+00,  0.16E+00,  0.29E+00,  0.4E+00,  0.48E+00,
           0.55E+00,  0.64E+00,  0.72E+00,  0.79E+00,  0.87E+00,  0.96E+00,
           0.104E+01,  0.112E+01, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
H       =  0.28E-01,  0.311E-01,  0.344E-01,  0.378E-01,  0.412E-01,
           0.454E-01,  0.54E-01,  0.704E-01,  0.104E+00,  0.18E+00,  0.25E+00,
           0.272E+00,  0.292E+00,  0.311E+00,  0.329E+00,  0.347E+00,
           0.365E+00,  0.383E+00,  0.402E+00,  0.411E+00,  0.499E+00, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I,
P       =  0.976E+00,  0.976E+00,  0.976E+00,  0.976E+00,  0.976E+00,
           0.976E+00,  0.976E+00,  0.976E+00,  0.976E+00,  0.976E+00,
           0.976E+00,  0.976E+00,  0.976E+00,  0.976E+00,  0.975E+00,
           0.974E+00,  0.971E+00,  0.968E+00,  0.963E+00,  0.954E+00,
           0.9416877E+00, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
DELTA   =  0.143E+01,
$END


$NAM3
AFOR    =  0.1E+01,  0.1E+01,  0.1E+01,  0.1E+01,  0.1E+01,  0.1E+01,
           0.99E+00,  0.98E+00,  0.96E+00,  0.9E+00,  0.66E+00,  0.2E+00,
           0.1E-01,  0.0,  0.0,  0.0,  0.0,  0.0,  0.0,  0.0,  0.0, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
          I, I, I, I, I, I, I, I, I, I, I, I, I, I, I,
$END
```

APPENDIX C

SAMPLE OUTPUT

In this appendix, the output for the calculation described in appendix B is given. First, identification of the case is printed out. After completion of the iterations the flow profiles are given, after which the equilibrium composition profiles are output by RATRAP. Only a sample of the composition profiles are shown here. Finally, the profile of the radiation flux vector is printed out. The output is given as follows:

```
                    VISCOUS STAGNATION STREAMLINE SOLUTION

                              WITH RADIATION

BODY RADIUS=  3.04800E+02 CM, FREESTREAM VELOCITY=  1.52500E+06 CM/SEC, DENSITY=  2.72000E-07 G/CC
 A SHOCK (NON DIM)= 1.22000,    BETA (NON DIM)= 2.50000,                 PRESSURE=  1.97000E+02 DYNES/SQ CM
 ABLATION RATE (NON DIM)= -.10000,   MASS FRACTIONS- HYDROGEN =0.00000,  OXYGEN=0.00000,  NITROGEN = .78000,  CARBON = .2200
 FREESTREAM COMPOSITION- N=    .78000,  O=    .22000,  C=  0.00000,  H=  0.00000


RANHUG SOLUTION

PSHOCK=  9.41689749E-01  ROSHOCK=  1.70585355E+01  VSHOCK=  5.86216795E-02  HSHOCK=  4.99371747E-01
```

DELTA= 1.43321101E+00 DELBA= 1.14968608E+01 REY= 9.48314311E+04

ETA(I)	Y(I)	QR(I)	CAPH(I)	V(I)	EMU(I)	PRAN(I)	YOYS(I)
0.	0.	-3.989709E-02	2.800013E-02	-5.122538E-04	6.436917E-01	7.079064E-01	0.
5.000000E-02	3.821030E-04	-4.012594E-02	3.081263E-02	-5.506918E-04	6.628098E-01	7.409187E-01	1.013016E-02
1.000000E-01	7.962958E-04	-4.037401E-02	3.425868E-02	-5.841457E-04	6.835926E-01	7.602514E-01	2.111106E-02
1.500000E-01	1.241459E-03	-4.070109E-02	3.842278E-02	-6.009434E-04	7.126751E-01	7.510083E-01	3.291303E-02
2.000000E-01	1.723479E-03	-4.105524E-02	4.298786E-02	-6.195919E-04	7.494184E-01	6.902536E-01	4.569216E-02
2.500000E-01	2.250103E-03	-4.134844E-02	4.877527E-02	-6.233564E-04	7.911803E-01	6.059531E-01	5.965378E-02
3.000000E-01	2.842980E-03	-4.167853E-02	5.889874E-02	-6.340208E-04	8.482358E-01	5.900988E-01	7.537191E-02
3.500000E-01	3.531493E-03	-4.238070E-02	7.694113E-02	-6.257030E-04	9.231293E-01	6.346786E-01	9.362548E-02
4.000000E-01	4.381608E-03	-4.324769E-02	1.120703E-01	-5.976021E-04	1.040424E+00	7.630011E-01	1.161634E-01
4.500000E-01	5.621403E-03	-4.459807E-02	1.879635E-01	-5.374347E-04	1.366311E+00	8.489653E-01	1.490323E-01
5.000000E-01	7.508546E-03	-4.665353E-02	2.534619E-01	2.334105E-04	1.598428E+00	3.619522E-01	1.990634E-01
5.500000E-01	9.849560E-03	-4.517633E-02	2.733345E-01	1.926409E-03	1.582374E+00	3.475065E-01	2.611274E-01
6.000000E-01	1.235369E-02	-4.359620E-02	2.957819E-01	4.306568E-03	1.550140E+00	3.358364E-01	3.275159E-01
6.500000E-01	1.500765E-02	-3.990416E-02	3.164136E-01	7.360373E-03	1.512331E+00	3.262837E-01	3.978765E-01
7.000000E-01	1.779896E-02	-3.602103E-02	3.351319E-01	1.111221E-02	1.473444E+00	3.187696E-01	4.718786E-01
7.500000E-01	2.072241E-02	-2.916608E-02	3.537244E-01	1.563643E-02	1.431549E+00	3.107011E-01	5.493838E-01
8.000000E-01	2.378199E-02	-2.199189E-02	3.720401E-01	2.098853E-02	1.386437E+00	3.018466E-01	6.304983E-01
8.500000E-01	2.698301E-02	-9.829270E-03	3.918516E-01	2.727595E-02	1.339113E+00	2.927529E-01	7.153624E-01
9.000000E-01	3.033711E-02	2.914981E-03	4.126163E-01	3.461759E-02	1.290905E+00	2.836595E-01	8.042848E-01
9.500000E-01	3.383703E-02	4.434637E-02	4.498601E-01	4.338700E-02	1.245466E+00	2.752380E-01	8.970733E-01
1.000000E+00	3.771936E-02	9.030465E-02	5.010900E-01	5.860377E-02	9.993658E-01	2.306848E-01	1.000000E+00

ETA(I)	ROV(I)	A(I)	P(I)	RO(I)	XNRO(I)	H(I)	XNH(I)
0.	-1.000000E-01	0.	9.761173E-01	1.951527E+02	1.951412E+02	2.800000E-02	2.800000E-02
5.000000E-02	-9.924396E-02	9.536205E-03	9.761136E-01	1.804490E+02	1.811262E+02	3.138136E-02	3.081248E-02
1.000000E-01	-9.706417E-02	2.005585E-02	9.761105E-01	1.662745E+02	1.666626E+02	3.458111E-02	3.425851E-02
1.500000E-01	-9.334795E-02	3.119411E-02	9.761085E-01	1.548980E+02	1.540167E+02	3.770567E-02	3.842260E-02
2.000000E-01	-8.792505E-02	4.347994E-02	9.761071E-01	1.411330E+02	1.394664E+02	4.144261E-02	4.298767E-02
2.500000E-01	-8.063055E-02	5.700017E-02	9.761064E-01	1.286055E+02	1.269916E+02	4.659978E-02	4.877507E-02
3.000000E-01	-7.127089E-02	7.230520E-02	9.761060E-01	1.119189E+02	1.108515E+02	5.617058E-02	5.889854E-02
3.500000E-01	-5.943349E-02	9.234868E-02	9.761065E-01	9.464081E+01	9.389961E+01	7.372470E-02	7.694093E-02
4.000000E-01	-4.457677E-02	1.151773E-01	9.761081E-01	7.437146E+01	7.390891E+01	1.086976E-01	1.120702E-01
4.500000E-01	-2.498915E-02	1.700667E-01	9.761166E-01	4.638199E+01	4.617058E+01	1.859130E-01	1.879634E-01
5.000000E-01	7.416170E-03	2.944836E-01	9.761195E-01	3.178005E+01	3.182610E+01	2.547815E-01	2.534619E-01
5.500000E-01	5.693602E-02	4.002848E-01	9.760655E-01	2.957343E+01	2.964103E+01	2.757263E-01	2.733327E-01
6.000000E-01	1.195418E-01	4.875373E-01	9.758536E-01	2.775983E+01	2.778787E+01	2.969440E-01	2.957726E-01
6.500000E-01	1.935969E-01	5.735127E-01	9.753752E-01	2.629487E+01	2.629757E+01	3.165518E-01	3.163865E-01
7.000000E-01	2.787872E-01	6.556514E-01	9.744824E-01	2.507536E+01	2.506592E+01	3.346064E-01	3.350702E-01
7.500000E-01	3.746239E-01	7.369009E-01	9.730016E-01	2.394328E+01	2.392861E+01	3.527941E-01	3.536022E-01
8.000000E-01	4.807406E-01	8.159726E-01	9.706987E-01	2.288966E+01	2.287455E+01	3.709259E-01	3.718199E-01
8.500000E-01	5.967372E-01	8.941515E-01	9.673068E-01	2.185935E+01	2.183653E+01	3.899858E-01	3.914796E-01
9.000000E-01	7.222385E-01	9.706369E-01	9.624048E-01	2.084001E+01	2.080885E+01	4.099460E-01	4.120171E-01
9.500000E-01	8.569591E-01	1.046138E+00	9.543970E-01	1.967499E+01	1.952243E+01	4.367558E-01	4.489189E-01
1.000000E+00	1.000000E+00	1.120000E+00	9.416897E-01	1.705990E+01	1.706198E+01	4.992441E-01	4.993728E-01

ETA(I)	T(I)	AMFC(I)	AMFN(I)	AMFO(I)	AMFH(I)	AFOR(I)	AONC(I)
0.	3.613975E+03	0.	7.800000E-01	2.200000E-01	0.	9.987159E-01	1.284070E-03
5.000000E-02	3.810559E+03	0.	7.800000E-01	2.200000E-01	0.	9.979070E-01	2.092989E-03
1.000000E-01	4.041588E+03	0.	7.800000E-01	2.200000E-01	0.	9.964582E-01	3.541793E-03
1.500000E-01	4.268651E+03	0.	7.800000E-01	2.200000E-01	0.	9.938745E-01	6.125505E-03
2.000000E-01	4.619094E+03	0.	7.800000E-01	2.200000E-01	0.	9.893142E-01	1.068581E-02
2.500000E-01	5.004771E+03	0.	7.800000E-01	2.200000E-01	0.	9.812354E-01	1.876461E-02
3.000000E-01	5.545945E+03	0.	7.800000E-01	2.200000E-01	0.	9.666189E-01	3.338105E-02
3.500000E-01	6.081784E+03	0.	7.800000E-01	2.200000E-01	0.	9.387366E-01	6.126344E-02
4.000000E-01	6.718330E+03	0.	7.800000E-01	2.200000E-01	0.	8.811335E-01	1.188665E-01
4.500000E-01	8.547436E+03	0.	7.800000E-01	2.200000E-01	0.	7.424072E-01	2.575928E-01
5.000000E-01	1.135178E+04	0.	7.800000E-01	2.200000E-01	0.	4.183161E-01	5.816839E-01
5.500000E-01	1.181809E+04	0.	7.800000E-01	2.200000E-01	0.	1.171349E-01	8.828651E-01
6.000000E-01	1.219760E+04	0.	7.800000E-01	2.200000E-01	0.	1.897675E-02	9.810233E-01
6.500000E-01	1.250590E+04	0.	7.800000E-01	2.200000E-01	0.	1.960880E-03	9.980391E-01
7.000000E-01	1.276497E+04	0.	7.800000E-01	2.200000E-01	0.	1.388550E-04	9.998611E-01
7.500000E-01	1.300704E+04	0.	7.800000E-01	2.200000E-01	0.	7.089309E-06	9.999929E-01
8.000000E-01	1.323332E+04	0.	7.800000E-01	2.200000E-01	0.	0.	1.000000E+00
8.500000E-01	1.345444E+04	0.	7.800000E-01	2.200000E-01	0.	0.	1.000000E+00
9.000000E-01	1.367395E+04	0.	7.800000E-01	2.200000E-01	0.	0.	1.000000E+00
9.500000E-01	1.391167E+04	0.	7.800000E-01	2.200000E-01	0.	0.	1.000000E+00
1.000000E+00	1.460291E+04	0.	7.800000E-01	2.200000E-01	0.	0.	1.000000E+00

```
 PATH LENGTH     PRESSURE    TEMPERATURE    ENTHALPY     DENSITY
0.              6.09388E-01  3.61398E+03  1.55566E+03  5.30459E-05
   MOLE FRACTIONS
    O+           O             N+           N            E            O2           N2
    CO           H2            C2           CN           C            C+           H
    C2H          C3H           C4H          HCN          C2H2         H+
0.              2.07175E-01  0.           3.99353E-04  0.           7.38121E-02  7.18614E-01
0.              0.           0.           0.           0.           0.           0.
0.              0.           0.           0.           0.           0.

 PATH LENGTH     PRESSURE    TEMPERATURE    ENTHALPY     DENSITY
1.01302E-02     6.09386E-01  3.79647E+03  1.74352E+03  4.92326E-05
   MOLE FRACTIONS
    O+           O             N+           N            E            O2           N2
    CO           H2            C2           CN           C            C+           H
    C2H          C3H           C4H          HCN          C2H2         H+
0.              2.51558E-01  0.           8.56953E-04  0.           4.71824E-02  7.00402E-01
0.              0.           0.           0.           0.           0.           0.
0.              0.           0.           0.           0.           0.

 PATH LENGTH     PRESSURE    TEMPERATURE    ENTHALPY     DENSITY
2.11111E-02     6.09384E-01  3.99909E+03  1.92130E+03  4.57768E-05
   MOLE FRACTIONS
    O+           O             N+           N            E            O2           N2
    CO           H2            C2           CN           C            C+           H
    C2H          C3H           C4H          HCN          C2H2         H+
0.              2.86503E-01  0.           1.85118E-03  0.           2.61533E-02  6.85493E-01
0.              0.           0.           0.           0.           0.           0.
0.              0.           0.           0.           0.           0.

 PATH LENGTH     PRESSURE    TEMPERATURE    ENTHALPY     DENSITY
3.29130E-02     6.09383E-01  4.29857E+03  2.09490E+03  4.18511E-05
   MOLE FRACTIONS
    O+           O             N+           N            E            O2           N2
    CO           H2            C2           CN           C            C+           H
    C2H          C3H           C4H          HCN          C2H2         H+
0.              3.12849E-01  0.           5.09863E-03  0.           1.00510E-02  6.72001E-01
0.              0.           0.           0.           0.           0.           0.
0.              0.           0.           0.           0.           0.

 PATH LENGTH     PRESSURE    TEMPERATURE    ENTHALPY     DENSITY
4.56922E-02     6.09382E-01  4.68828E+03  2.30252E+03  3.78972E-05
   MOLE FRACTIONS
    O+           O             N+           N            E            O2           N2
    CO           H2            C2           CN           C            C+           H
    C2H          C3H           C4H          HCN          C2H2         H+
0.              3.22968E-01  0.           1.58038E-02  0.           2.93056E-03  6.58298E-01
0.              0.           0.           0.           0.           0.           0.
0.              0.           0.           0.           0.           0.
```

```
TOTAL QMINUS =  0.
TOTAL QPLUS  = 20.7296E+02


TOTAL QMINUS = 56.8029E-09
TOTAL QPLUS  = 21.0752E+02


TOTAL QMINUS = 29.7649E-07
TOTAL QPLUS  = 21.2869E+02


TOTAL QMINUS = 16.8581E-05
TOTAL QPLUS  = 21.0812E+02


TOTAL QMINUS = 11.5100E-03
TOTAL QPLUS  = 20.6813E+02


TOTAL QMINUS = 77.6169E+00
TOTAL QPLUS  = 22.1093E+02


TOTAL QMINUS = 54.7567E+01
TOTAL QPLUS  = 24.8172E+02


TOTAL QMINUS = 11.5258E+02
TOTAL QPLUS  = 27.2646E+02


TOTAL QMINUS = 18.3072E+02
TOTAL QPLUS  = 28.2948E+02


TOTAL QMINUS = 26.9051E+02
TOTAL QPLUS  = 27.4797E+02


TOTAL QMINUS = 41.3077E+02
TOTAL QPLUS  =  0.
```

ETA(I)	Y(I)	YOYS(I)	QRAD(I),W/SQ CM
0.	0.	0.	-3.882058E+03
5.000000E-02	3.821030E-04	1.013016E-02	-3.909761E+03
1.000000E-01	7.962958E-04	2.111106E-02	-3.936354E+03
1.500000E-01	1.241459E-03	3.291303E-02	-3.960952E+03
2.000000E-01	1.723479E-03	4.569216E-02	-3.982931E+03
2.500000E-01	2.250103E-03	5.965378E-02	-4.004473E+03
3.000000E-01	2.842980E-03	7.537191E-02	-4.024465E+03
3.500000E-01	3.531493E-03	9.362548E-02	-4.076455E+03
4.000000E-01	4.381608E-03	1.161634E-01	-4.168434E+03
4.500000E-01	5.621403E-03	1.490323E-01	-4.294697E+03
5.000000E-01	7.508546E-03	1.990634E-01	-4.504456E+03
5.500000E-01	9.849560E-03	2.611274E-01	-4.497930E+03
6.000000E-01	1.235369E-02	3.275159E-01	-4.246319E+03
6.500000E-01	1.500765E-02	3.978765E-01	-3.947319E+03
7.000000E-01	1.779896E-02	4.718786E-01	-3.512482E+03
7.500000E-01	2.072241E-02	5.493838E-01	-2.925811E+03
8.000000E-01	2.378199E-02	6.304983E-01	-2.167645E+03
8.500000E-01	2.698301E-02	7.153624E-01	-1.123870E+03
9.000000E-01	3.033711E-02	8.042848E-01	2.123313E+02
9.500000E-01	3.383703E-02	8.970733E-01	3.505130E+03
1.000000E+00	3.771936E-02	1.000000E+00	8.804495E+03

APPENDIX D

LANGLEY LIBRARY SUBROUTINE ITR1

Language: FORTRAN

Purpose: To solve the single equation of the form $x = f(x)$ for one real root by the Newton-Raphson iteration method.

Use: CALL ITR1 (X, DELTX, FOFX, E1, E2, MAXI, ICODE)

X	An initial guess supplied by the user. On a normal return to the calling program from ITR1, X contains the root.
DELTX	An increment supplied by the user so that $\frac{f(x + \text{DELTX}) - f(x)}{\text{DELTX}}$ is a reasonable approximation to the derivative of $f(x)$.
FOFX	A function subprogram to evaluate $f(x)$.
E1	Relative error criterion.
E2	Absolute error criterion.
MAXI	A maximum iteration count supplied by the user.
ICODE	An integer supplied by ITR1 as an error code. This code should be tested by the user on return to the calling program. ICODE = 0: Normal return. ICODE = 1: Maximum iteration exceeded. ICODE = 2: Derivative = 0.

Restrictions: A function subprogram with a single argument x must be written by the user to evaluate $f(x)$. The name given to the FOFX subprogram must appear in an EXTERNAL statement in the calling program.

Method: The Newton-Raphson iteration technique (ref. (a) of this subroutine) is used where

$$x_{n+1} = q_n + (1 - q)\ f(x_n)$$

$$q = \frac{a}{a - 1}$$

$$a = \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}$$

Accuracy: The iteration process is continued until either of two convergence criteria are satisfied. These criteria are given as follows:

If

$$\left|f(x_n)\right| \geqq \epsilon_1$$

then

$$\left|\frac{f(x_n) - x_n}{f(x_n)}\right| \leqq \epsilon_1 \tag{1}$$

and if

$$\left|f(x_n)\right| < \epsilon_1$$

then

$$\left|f(x_n) - x_n\right| \leqq \epsilon_2 \tag{2}$$

Reference: (a) Scarborough, James B.: Numerical Mathematical Analysis, Fourth ed., Johns Hopkins Press, 1958, p. 192.

Storage: 137_8 locations.

Subroutine date: August 1, 1968.

APPENDIX E

LANGLEY LIBRARY SUBROUTINE FTLUP

Language: FORTRAN

Purpose: Computes $y = F(x)$ from a table of values using first or second order interpolation. An option to give y a constant value for any x is also provided.

Use:	CALL FTLUP (X, Y, M, N, VARI, VARD)
X	The name of the independent variable x
Y	The name of the dependent variable $y = f(x)$
M	The order of interpolation (an integer) M = 0 for y a constant as explained in the NOTE below. M = 1 or 2. First or second order if VARI is strictly increasing (not equal). M = -1 or -2. First or second order if VARI is strictly decreasing (not equal).
N	The number of points in the table (an integer).
VARI	The name of a one-dimensional array which contains the N values of the independent variable.
VARD	The name of a one-dimensional array which contains the N values of the dependent variable.
Note:	VARD(I) corresponds to VARI(I) for $I = 1, 2, \ldots, N$. For $M = 0$ or $N \leq 1$, $y = F(VARI(1))$ for any value of x. The program extrapolates.

Restrictions: All the numbers must be floating point. The values of the independent variable x in the table must be strictly increasing or strictly decreasing. The following arrays must be dimensioned by the calling program as indicated: VARI(N), VARD(N).

Accuracy: A function of the order of interpolation used.

References: (a) Nielson, Kaj L.: Methods in Numerical Analysis. The Macmillan Co., c.1956, pp. 87-91.

(b) Milne, William Edmund: Numerical Calculus. Princeton Univ. Press, c.1949, pp. 69-73.

Storage: 430_8 locations.

Error condition: If the VARI values are not in order, the subroutine will print "TABLE BELOW OUT OF ORDER FOR FTLUP AT POSITION xxx TABLE IS STORED IN LOCATION xxxxxx" (absolute). It then prints the contents of VARI and VARD and stops the program.

Subroutine date: September 12, 1969.

APPENDIX F

LANGLEY LIBRARY SUBROUTINE DISCOT

Language: FORTRAN

Purpose: DISCOT performs single or double interpolation for continuous or discontinuous functions. Given a table of some function y with two independent variables, x and z, this subroutine performs K_xth- and K_zth-order interpolation to calculate the dependent variable. In this subroutine all single-line functions are read in as two separate arrays and all multiline functions are read in as three separate arrays; that is,

$$x_i \qquad (i = 1, 2, \ldots, L)$$

$$y_j \qquad (j = 1, 2, \ldots, M)$$

$$z_k \qquad (k = 1, 2, \ldots, N)$$

Use: CALL DISCOT (XA, ZA, TABX, TABY, TABZ, NC, NY, NZ, ANS)

XA	The x argument
ZA	The z argument (may be the same name as x on single lines)
TABX	A one-dimensional array of x values
TABY	A one-dimensional array of y values
TABZ	A one-dimensional array of z values
NC	A control word that consists of a sign (+ or -) and three digits. The control word is formed as follows:

(1) If NX = NY, the sign is negative. If NX ≠ NY, then NX is computed by DISCOT as NX = NY/Nz, and the sign is positive and may be omitted if desired.

(2) A one in the hundreds position of the word indicates that no extrapolation occurs above z_{max}. With a zero in this position, extrapolation occurs when $z > z_{max}$. The zero may be omitted if desired.

(3) A digit (1 to 7) in the tens position of the word indicates the order of interpolation in the x-direction.

(4) A digit (1 to 7) in the units position of the word indicates the order of interpolation in the z-direction.

NY	The number of points in y array
NZ	The number of points in z array
ANS	The dependent variable y

The following programs will illustrate various ways to use DISCOT:

CASE I:

Given $y = f(x)$
NY = 50
NX (number of points in x array) = NY
Extrapolation when $z > z_{max}$
Second-order interpolation in x-direction
No interpolation in z-direction
Control word = -020

```
      DIMENSION TABX (50), TABY (50)
    1 FORMAT (8E 9.5)
      READ (5,1) TABX, TABY
      READ (5,1) XA
      CALL DISCOT (XA, XA, TABX, TABY, TABY, -020, 50, 0, ANS)
```

CASE II:

Given $y = f(x,z)$
NY = 800
NZ = 10
NX = NY/NZ (computed by DISCOT)
Extrapolation when $z > z_{max}$
Linear interpolation in x-direction
Linear interpolation in z-direction
Control word = 11

```
      DIMENSION TABX (800), TABY (800), TABZ (10)
    1 FORMAT (8E 9.5)
      READ (5,1) TABX, TABY, TABZ
      READ (5,1) XA, ZA
      CALL DISCOT (XA, ZA, TABX, TABY, TABZ, 11, 800, 10, ANS)
```

CASE III:

Given $y = f(x,z)$
NY = 800
NZ = 10
NX = NY
Extrapolation when $z > z_{max}$
Seventh-order interpolation in x-direction
Third-order interpolation in z-direction
Control word = -73

```
      DIMENSION TABX (800), TABY (800), TABZ (10)
    1 FORMAT (8E 9.5)
      READ (5,1) TABX, TABY, TABZ
      READ (5,1) XA, ZA
      CALL DISCOT (XA, ZA, TABX, TABY, TABZ, -73, 800, 10, ANS)
```

CASE IV:

Same as Case III with no extrapolation above z_{max}. Control word = -173

```
      CALL DISCOT (XA, ZA, TABX, TABY, TABZ, -173, 800, 10, ANS)
```

Restrictions: See rule (5c) of section "Method" for restrictions on tabulating arrays and discontinuous functions. The order of interpolation in the x- and z-directions may be from 1 to 7. The following subprograms are used by DISCOT: UNS, DISSER, LAGRAN.

Method: Lagrange's interpolation formula is used in both the x- and z-directions for interpolation. This method is explained in detail in reference (a) of this subroutine. For a search in either the x- or z-direction, the following rules are observed:

(1) If $x < x_1$, the routine chooses the following points for extrapolation:

$$x_1, x_2, \ldots, x_{k+1} \quad \text{and} \quad y_1, y_2, \ldots, y_{k+1}$$

(2) If $x > x_n$, the routine chooses the following points for extrapolation:

$$x_{n-k}, x_{n-k+1}, \ldots, x_n \quad \text{and} \quad y_{n-k}, y_{n-k+1}, \ldots, y_n$$

(3) If $x \leqq x_n$, the routine chooses the following points for interpolation:

When k is odd,

$$x_{i-\frac{k+1}{2}}, x_{i-\frac{k+1}{2}+1}, \ldots, x_{i-\frac{k+1}{2}+k} \quad \text{and} \quad y_{i-\frac{k+1}{2}}, y_{i-\frac{k+1}{2}+1}, \ldots, y_{i-\frac{k+1}{2}+k}$$

When k is even,

$$x_{i-\frac{k}{2}}, x_{i-\frac{k}{2}+1}, \ldots, x_{i-\frac{k}{2}+k} \quad \text{and} \quad y_{i-\frac{k}{2}}, y_{i-\frac{k}{2}+1}, \ldots, y_{i-\frac{k}{2}+k}$$

(4) If any of the subscripts in rule (3) become negative or greater than n (number of points), rules (1) and (2) apply. When discontinuous functions are tabulated, the independent variable at the point of discontinuity is repeated.

(5) The subroutine will automatically examine the points selected before interpolation and if there is a discontinuity, the following rules apply. Let x_d and x_{d+1} be the point of discontinuity.

(a) If $x \leqq x_d$, points previously chosen are modified for interpolation as shown:

$$x_{d-k}, x_{d-k+1}, \ldots, x_d \quad \text{and} \quad y_{d-k}, y_{d-k+1}, \ldots, y_d$$

(b) If $x > x_d$, points previously chosen are modified for interpolation as shown:

$$x_{d+1}, x_{d+2}, \ldots, x_{d+k} \quad \text{and} \quad y_{d+1}, y_{d+2}, \ldots, y_{d+k}$$

(c) When tabulating discontinuous functions, there must always be $k + 1$ points above and below the discontinuity in order to get proper interpolation.

(6) When tabulating arrays for this subroutine, both independent variables must be in ascending order.

(7) In some engineering programs with many tables, it is quite desirable to read in one array of x values that could be used for all lines of a multiline function or different functions. Even though this situation is not always applicable, the subroutine has been written to handle it. This procedure not only saves much time in preparing tabular data, but also can save many locations previously used when every y coordinate had to have a corresponding x coordinate. Another additional feature that may be useful is the possibility of a multiline function with no extrapolation above the top line.

Accuracy: A function of the order of interpolation used.

Reference: (a) Nielsen, Kaj L.: Methods in Numerical Analysis. The Macmillan Co., c.1956.

Storage: 555_8 locations.

Subprograms used: UNS 40_8 locations.
DISSER 110_8 locations.
LAGRAN 55_8 locations.

Subroutine date: August 1, 1968.

REFERENCES

1. Garrett, Lloyd Bernard: An Implicit Finite Difference Solution to the Viscous Radiating Shock Layer With Strong Blowing. Ph. D. Diss., North Carolina State Univ. at Raleigh, 1971. (Available as NASA TM X-67584.)
2. Garrett, L. Bernard; Smith, G. Louis; and Perkins, John N.: An Implicit Finite-Difference Solution to the Viscous Shock Layer, Including the Effects of Radiation and Strong Blowing. NASA TR-388, 1972.
3. Wilson, K. H.: RATRAP – A Radiation Transport Code. 6-77-67-12, Lockheed Missiles & Space Co., Mar. 14, 1967.
4. Hansen, C. Frederick: Approximations for the Thermodynamic and Transport Properties of High-Temperature Air. NASA TR R-50, 1959. (Supersedes NACA TN 4150.)

NASA-Langley, 1973 —— 12 L-8918

"The aeronautical and space activities of the United States shall be conducted so as to contribute . . . to the expansion of human knowledge of phenomena in the atmosphere and space. The Administration shall provide for the widest practicable and appropriate dissemination of information concerning its activities and the results thereof."

—NATIONAL AERONAUTICS AND SPACE ACT OF 1958

NASA SCIENTIFIC AND TECHNICAL PUBLICATIONS

TECHNICAL REPORTS: Scientific and technical information considered important, complete, and a lasting contribution to existing knowledge.

TECHNICAL NOTES: Information less broad in scope but nevertheless of importance as a contribution to existing knowledge.

TECHNICAL MEMORANDUMS: Information receiving limited distribution because of preliminary data, security classification, or other reasons. Also includes conference proceedings with either limited or unlimited distribution.

CONTRACTOR REPORTS: Scientific and technical information generated under a NASA contract or grant and considered an important contribution to existing knowledge.

TECHNICAL TRANSLATIONS: Information published in a foreign language considered to merit NASA distribution in English.

SPECIAL PUBLICATIONS: Information derived from or of value to NASA activities. Publications include final reports of major projects, monographs, data compilations, handbooks, sourcebooks, and special bibliographies.

TECHNOLOGY UTILIZATION PUBLICATIONS: Information on technology used by NASA that may be of particular interest in commercial and other non-aerospace applications. Publications include Tech Briefs, Technology Utilization Reports and Technology Surveys.

Details on the availability of these publications may be obtained from:

SCIENTIFIC AND TECHNICAL INFORMATION OFFICE

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

Washington, D.C. 20546